## CONTENTS

<table>
<thead>
<tr>
<th>Apex Reference Guide</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release Notes</td>
<td>3</td>
</tr>
<tr>
<td>Apex DML Operations</td>
<td>4</td>
</tr>
<tr>
<td>Apex DML Statements</td>
<td>4</td>
</tr>
<tr>
<td>ApexPages Namespace</td>
<td>8</td>
</tr>
<tr>
<td>Action Class</td>
<td>8</td>
</tr>
<tr>
<td>Component Class</td>
<td>10</td>
</tr>
<tr>
<td>IdeaStandardController Class</td>
<td>12</td>
</tr>
<tr>
<td>IdeaStandardSetController Class</td>
<td>14</td>
</tr>
<tr>
<td>KnowledgeArticleVersionStandardController Class</td>
<td>18</td>
</tr>
<tr>
<td>Message Class</td>
<td>21</td>
</tr>
<tr>
<td>StandardController Class</td>
<td>25</td>
</tr>
<tr>
<td>StandardSetController Class</td>
<td>30</td>
</tr>
<tr>
<td>Applauncher Namespace</td>
<td>39</td>
</tr>
<tr>
<td>AppMenu Class</td>
<td>40</td>
</tr>
<tr>
<td>ChangePasswordController Class</td>
<td>42</td>
</tr>
<tr>
<td>CommunityLogoController Class</td>
<td>42</td>
</tr>
<tr>
<td>EmployeeLoginLinkController Class</td>
<td>42</td>
</tr>
<tr>
<td>ForgotPasswordController Class</td>
<td>42</td>
</tr>
<tr>
<td>IdentityHeaderController Class</td>
<td>42</td>
</tr>
<tr>
<td>LoginFormController Class</td>
<td>43</td>
</tr>
<tr>
<td>SelfRegisterController Class</td>
<td>43</td>
</tr>
<tr>
<td>SocialLoginController Class</td>
<td>43</td>
</tr>
<tr>
<td>Approval Namespace</td>
<td>43</td>
</tr>
<tr>
<td>LockResult Class</td>
<td>44</td>
</tr>
<tr>
<td>ProcessRequest Class</td>
<td>46</td>
</tr>
<tr>
<td>ProcessResult Class</td>
<td>47</td>
</tr>
<tr>
<td>ProcessSubmitRequest Class</td>
<td>50</td>
</tr>
<tr>
<td>ProcessWorkitemRequest Class</td>
<td>54</td>
</tr>
<tr>
<td>UnlockResult Class</td>
<td>56</td>
</tr>
<tr>
<td>Auth Namespace</td>
<td>58</td>
</tr>
<tr>
<td>AuthConfiguration Class</td>
<td>60</td>
</tr>
<tr>
<td>AuthProviderCallbackState Class</td>
<td>71</td>
</tr>
<tr>
<td>AuthProviderPlugin Interface</td>
<td>73</td>
</tr>
<tr>
<td>AuthProviderPluginClass</td>
<td>76</td>
</tr>
<tr>
<td>AuthProviderTokenResponse Class</td>
<td>86</td>
</tr>
<tr>
<td>AuthToken Class</td>
<td>89</td>
</tr>
<tr>
<td>CommunitiesUtil Class</td>
<td>93</td>
</tr>
<tr>
<td>ConfigurableSelfRegHandler Interface</td>
<td>95</td>
</tr>
<tr>
<td>Class/Interface/Enum</td>
<td>Page</td>
</tr>
<tr>
<td>---------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>ConfirmUserRegistrationHandler Interface</td>
<td>100</td>
</tr>
<tr>
<td>ConnectedAppPlugin Class</td>
<td>103</td>
</tr>
<tr>
<td>HeadlessSelfRegistrationHandler Interface</td>
<td>109</td>
</tr>
<tr>
<td>InvocationContext Enum</td>
<td>112</td>
</tr>
<tr>
<td>JWS Class</td>
<td>113</td>
</tr>
<tr>
<td>JWT Class</td>
<td>115</td>
</tr>
<tr>
<td>JWTBearerTokenExchange Class</td>
<td>120</td>
</tr>
<tr>
<td>LightningLoginEligibility Enum</td>
<td>126</td>
</tr>
<tr>
<td>LoginDiscoveryHandler Interface</td>
<td>127</td>
</tr>
<tr>
<td>LoginDiscoveryMethod Enum</td>
<td>134</td>
</tr>
<tr>
<td>MyDomainLoginDiscoveryHandler Interface</td>
<td>134</td>
</tr>
<tr>
<td>OAuthRefreshResult Class</td>
<td>138</td>
</tr>
<tr>
<td>RegistrationHandler Interface</td>
<td>141</td>
</tr>
<tr>
<td>SamUtilHandler Interface</td>
<td>146</td>
</tr>
<tr>
<td>SessionManagement Class</td>
<td>150</td>
</tr>
<tr>
<td>SessionLevel Enum</td>
<td>162</td>
</tr>
<tr>
<td>UserData Class</td>
<td>163</td>
</tr>
<tr>
<td>VerificationMethod Enum</td>
<td>168</td>
</tr>
<tr>
<td>VerificationPolicy Enum</td>
<td>168</td>
</tr>
<tr>
<td>VerificationResult Class</td>
<td>169</td>
</tr>
<tr>
<td>Auth Exceptions</td>
<td>172</td>
</tr>
<tr>
<td>Cache Namespace</td>
<td>173</td>
</tr>
<tr>
<td>CacheBuilderInterface</td>
<td>174</td>
</tr>
<tr>
<td>Org Class</td>
<td>176</td>
</tr>
<tr>
<td>OrgPartition Class</td>
<td>193</td>
</tr>
<tr>
<td>Partition Class</td>
<td>196</td>
</tr>
<tr>
<td>Session Class</td>
<td>211</td>
</tr>
<tr>
<td>SessionPartition Class</td>
<td>228</td>
</tr>
<tr>
<td>Cache Exceptions</td>
<td>231</td>
</tr>
<tr>
<td>Visibility Enum</td>
<td>232</td>
</tr>
<tr>
<td>Canvas Namespace</td>
<td>232</td>
</tr>
<tr>
<td>ApplicationContext Interface</td>
<td>233</td>
</tr>
<tr>
<td>CanvasLifecycleHandler Interface</td>
<td>236</td>
</tr>
<tr>
<td>ContextTypeEnum Enum</td>
<td>238</td>
</tr>
<tr>
<td>EnvironmentContext Interface</td>
<td>239</td>
</tr>
<tr>
<td>RenderContext Interface</td>
<td>245</td>
</tr>
<tr>
<td>Test Class</td>
<td>247</td>
</tr>
<tr>
<td>Canvas Exceptions</td>
<td>250</td>
</tr>
<tr>
<td>ChatterAnswers Namespace</td>
<td>251</td>
</tr>
<tr>
<td>AccountCreator Interface</td>
<td>251</td>
</tr>
<tr>
<td>CommercePayments Namespace</td>
<td>253</td>
</tr>
<tr>
<td>AbstractResponse Class</td>
<td>256</td>
</tr>
<tr>
<td>AddressRequest Class</td>
<td>259</td>
</tr>
<tr>
<td>AuditParamsRequest Class</td>
<td>263</td>
</tr>
<tr>
<td>Class/Enum Name</td>
<td>Page</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>AbstractTransactionResponse Class</td>
<td>265</td>
</tr>
<tr>
<td>AuthApiPaymentMethodRequest Class</td>
<td>270</td>
</tr>
<tr>
<td>AuthorizationReversalRequest Class</td>
<td>271</td>
</tr>
<tr>
<td>AuthorizationReversalResponse Class</td>
<td>275</td>
</tr>
<tr>
<td>AuthorizationRequest Class</td>
<td>280</td>
</tr>
<tr>
<td>AuthorizationResponse Class</td>
<td>285</td>
</tr>
<tr>
<td>BaseApiPaymentMethodRequest Class</td>
<td>292</td>
</tr>
<tr>
<td>BaseNotification Class</td>
<td>295</td>
</tr>
<tr>
<td>BasePaymentMethodRequest Class</td>
<td>300</td>
</tr>
<tr>
<td>BaseRequest Class</td>
<td>302</td>
</tr>
<tr>
<td>CaptureNotification Class</td>
<td>303</td>
</tr>
<tr>
<td>CaptureRequest Class</td>
<td>308</td>
</tr>
<tr>
<td>CaptureResponse Class</td>
<td>309</td>
</tr>
<tr>
<td>CardCategory Enum</td>
<td>314</td>
</tr>
<tr>
<td>CardPaymentMethodRequest Class</td>
<td>315</td>
</tr>
<tr>
<td>CustomMetadataTypeInfo Class</td>
<td>321</td>
</tr>
<tr>
<td>GatewayErrorResponse Class</td>
<td>322</td>
</tr>
<tr>
<td>GatewayNotificationResponse Class</td>
<td>324</td>
</tr>
<tr>
<td>GatewayResponse Interface</td>
<td>326</td>
</tr>
<tr>
<td>NotificationClient Class</td>
<td>327</td>
</tr>
<tr>
<td>NotificationSaveResult Class</td>
<td>328</td>
</tr>
<tr>
<td>NotificationStatus Enum</td>
<td>330</td>
</tr>
<tr>
<td>PaymentGatewayAdapter Interface</td>
<td>330</td>
</tr>
<tr>
<td>PaymentGatewayAsyncAdapter Interface</td>
<td>331</td>
</tr>
<tr>
<td>PaymentGatewayContext Class</td>
<td>334</td>
</tr>
<tr>
<td>PaymentGatewayNotificationContext Class</td>
<td>336</td>
</tr>
<tr>
<td>PaymentMethodTokenizationRequest Class</td>
<td>337</td>
</tr>
<tr>
<td>PaymentMethodTokenizationResponse Class</td>
<td>342</td>
</tr>
<tr>
<td>PaymentGatewayNotificationRequest Class</td>
<td>348</td>
</tr>
<tr>
<td>PaymentsHttp Class</td>
<td>350</td>
</tr>
<tr>
<td>RefundRequest Class</td>
<td>351</td>
</tr>
<tr>
<td>ReferencedRefundNotification Class</td>
<td>353</td>
</tr>
<tr>
<td>ReferencedRefundRequest</td>
<td>357</td>
</tr>
<tr>
<td>ReferencedRefundResponse Class</td>
<td>359</td>
</tr>
<tr>
<td>RequestType Enum</td>
<td>364</td>
</tr>
<tr>
<td>SaleApiPaymentMethodRequest Class</td>
<td>364</td>
</tr>
<tr>
<td>SaleRequest Class</td>
<td>367</td>
</tr>
<tr>
<td>SaleResponse Class</td>
<td>371</td>
</tr>
<tr>
<td>SalesforceResultCode Enum</td>
<td>376</td>
</tr>
<tr>
<td>SalesforceResultCodeInfo</td>
<td>377</td>
</tr>
<tr>
<td>ConnectApi Namespace</td>
<td>378</td>
</tr>
<tr>
<td>ActionLinks Class</td>
<td>382</td>
</tr>
<tr>
<td>Announcements Class</td>
<td>391</td>
</tr>
<tr>
<td>BotVersionActivation Class</td>
<td>396</td>
</tr>
<tr>
<td>Class</td>
<td>Page</td>
</tr>
<tr>
<td>----------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>CdpCalculatedInsight Class</td>
<td>398</td>
</tr>
<tr>
<td>CdpIdentityResolution Class</td>
<td>404</td>
</tr>
<tr>
<td>CdpQuery Class</td>
<td>407</td>
</tr>
<tr>
<td>CdpSegment Class</td>
<td>424</td>
</tr>
<tr>
<td>Chatter Class</td>
<td>431</td>
</tr>
<tr>
<td>ChatterFavorites Class</td>
<td>436</td>
</tr>
<tr>
<td>ChatterFeeds Class</td>
<td>458</td>
</tr>
<tr>
<td>ChatterGroups Class</td>
<td>861</td>
</tr>
<tr>
<td>ChatterMessages Class</td>
<td>906</td>
</tr>
<tr>
<td>ChatterUsers Class</td>
<td>930</td>
</tr>
<tr>
<td>Clm Class</td>
<td>965</td>
</tr>
<tr>
<td>CommerceBuyerExperience Class</td>
<td>967</td>
</tr>
<tr>
<td>CommerceCart Class</td>
<td>1025</td>
</tr>
<tr>
<td>CommerceCatalog Class</td>
<td>1062</td>
</tr>
<tr>
<td>CommercePromotions Class</td>
<td>1079</td>
</tr>
<tr>
<td>CommerceSearch Class</td>
<td>1081</td>
</tr>
<tr>
<td>CommerceSearchConnectFamily Class</td>
<td>1085</td>
</tr>
<tr>
<td>CommerceSearchSettings Class</td>
<td>1087</td>
</tr>
<tr>
<td>CommerceStorePricing Class</td>
<td>1090</td>
</tr>
<tr>
<td>CommerceWishlist Class</td>
<td>1094</td>
</tr>
<tr>
<td>Communities Class</td>
<td>1112</td>
</tr>
<tr>
<td>CommunityModeration Class</td>
<td>1115</td>
</tr>
<tr>
<td>ContentHub Class</td>
<td>1144</td>
</tr>
<tr>
<td>ConversationApplicationDefinition Class</td>
<td>1218</td>
</tr>
<tr>
<td>Datacloud Class</td>
<td>1219</td>
</tr>
<tr>
<td>EmailMergeFieldService Class</td>
<td>1224</td>
</tr>
<tr>
<td>EmployeeProfiles Class</td>
<td>1225</td>
</tr>
<tr>
<td>ExternalEmailServices Class</td>
<td>1234</td>
</tr>
<tr>
<td>ExternalManagedAccount Class</td>
<td>1235</td>
</tr>
<tr>
<td>FieldService Class</td>
<td>1238</td>
</tr>
<tr>
<td>FulfillmentOrder Class</td>
<td>1241</td>
</tr>
<tr>
<td>Knowledge Class</td>
<td>1246</td>
</tr>
<tr>
<td>LightningScheduler Class</td>
<td>1251</td>
</tr>
<tr>
<td>ManagedContent Class</td>
<td>1256</td>
</tr>
<tr>
<td>ManagedContentDelivery Class</td>
<td>1277</td>
</tr>
<tr>
<td>ManagedTopics Class</td>
<td>1294</td>
</tr>
<tr>
<td>MarketingIntegration Class</td>
<td>1309</td>
</tr>
<tr>
<td>Mentions Class</td>
<td>1311</td>
</tr>
<tr>
<td>Missions Class</td>
<td>1317</td>
</tr>
<tr>
<td>NamedCredentials Class</td>
<td>1322</td>
</tr>
<tr>
<td>NavigationMenu Class</td>
<td>1332</td>
</tr>
<tr>
<td>NextBestAction Class</td>
<td>1336</td>
</tr>
<tr>
<td>OmnichannelInventoryService Class</td>
<td>1342</td>
</tr>
<tr>
<td>Orchestration Class</td>
<td>1349</td>
</tr>
<tr>
<td>Class Name</td>
<td>Page</td>
</tr>
<tr>
<td>------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>OrderPaymentSummary Class</td>
<td>1350</td>
</tr>
<tr>
<td>OrderSummary Class</td>
<td>1352</td>
</tr>
<tr>
<td>OrderSummaryCreation Class</td>
<td>1364</td>
</tr>
<tr>
<td>Organization Class</td>
<td>1365</td>
</tr>
<tr>
<td>PardotBusinessUnitContext Class</td>
<td>1365</td>
</tr>
<tr>
<td>Payments Class</td>
<td>1367</td>
</tr>
<tr>
<td>Personalization Class</td>
<td>1372</td>
</tr>
<tr>
<td>PickTicket Class</td>
<td>1385</td>
</tr>
<tr>
<td>QuestionAndAnswers Class</td>
<td>1386</td>
</tr>
<tr>
<td>Recommendations Class</td>
<td>1390</td>
</tr>
<tr>
<td>Records Class</td>
<td>1450</td>
</tr>
<tr>
<td>Repricing Class</td>
<td>1452</td>
</tr>
<tr>
<td>ReturnOrder Class</td>
<td>1455</td>
</tr>
<tr>
<td>Routing Class</td>
<td>1457</td>
</tr>
<tr>
<td>SalesforceInbox Class</td>
<td>1462</td>
</tr>
<tr>
<td>Sites Class</td>
<td>1463</td>
</tr>
<tr>
<td>SmartDataDiscovery Class</td>
<td>1464</td>
</tr>
<tr>
<td>SocialEngagement Class</td>
<td>1464</td>
</tr>
<tr>
<td>Surveys Class</td>
<td>1475</td>
</tr>
<tr>
<td>TaxPlatform Class</td>
<td>1476</td>
</tr>
<tr>
<td>Topics Class</td>
<td>1477</td>
</tr>
<tr>
<td>UserProfiles Class</td>
<td>1514</td>
</tr>
<tr>
<td>Zones Class</td>
<td>1524</td>
</tr>
<tr>
<td>ConnectApi Input Classes</td>
<td>1534</td>
</tr>
<tr>
<td>ConnectApi Output Classes</td>
<td>1668</td>
</tr>
<tr>
<td>ConnectApi Enums</td>
<td>2011</td>
</tr>
<tr>
<td>ConnectApi Exceptions</td>
<td>2043</td>
</tr>
<tr>
<td>ConnectApi Utilities</td>
<td>2043</td>
</tr>
<tr>
<td>ConnectApi Release Notes</td>
<td>2043</td>
</tr>
<tr>
<td>Database Namespace</td>
<td>2044</td>
</tr>
<tr>
<td>Batchable Interface</td>
<td>2045</td>
</tr>
<tr>
<td>BatchableContext Interface</td>
<td>2047</td>
</tr>
<tr>
<td>DeletedRecord Class</td>
<td>2048</td>
</tr>
<tr>
<td>DeleteResult Class</td>
<td>2049</td>
</tr>
<tr>
<td>DMLOptions Class</td>
<td>2051</td>
</tr>
<tr>
<td>DMLOptions.AssignmentRuleHeader Class</td>
<td>2054</td>
</tr>
<tr>
<td>DMLOptions.DuplicateRuleHeader Class</td>
<td>2055</td>
</tr>
<tr>
<td>DMLOptions.EmailHeader Class</td>
<td>2057</td>
</tr>
<tr>
<td>DuplicateError Class</td>
<td>2060</td>
</tr>
<tr>
<td>EmptyRecycleBinResult Class</td>
<td>2062</td>
</tr>
<tr>
<td>Error Class</td>
<td>2063</td>
</tr>
<tr>
<td>GetDeletedResult Class</td>
<td>2065</td>
</tr>
<tr>
<td>GetUpdatedResult Class</td>
<td>2066</td>
</tr>
<tr>
<td>LeadConvert Class</td>
<td>2067</td>
</tr>
<tr>
<td>Class/Interface/Enum</td>
<td>Page</td>
</tr>
<tr>
<td>----------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>TableSelection Class</td>
<td>2173</td>
</tr>
<tr>
<td>UpsertContext Class</td>
<td>2175</td>
</tr>
<tr>
<td>UpsertResult Class</td>
<td>2176</td>
</tr>
<tr>
<td>DataSource Exceptions</td>
<td>2179</td>
</tr>
<tr>
<td>DataWeave Namespace</td>
<td>2179</td>
</tr>
<tr>
<td>Result Class</td>
<td>2180</td>
</tr>
<tr>
<td>Script Class</td>
<td>2181</td>
</tr>
<tr>
<td>Dom Namespace</td>
<td>2183</td>
</tr>
<tr>
<td>Document Class</td>
<td>2184</td>
</tr>
<tr>
<td>XmlNode Class</td>
<td>2186</td>
</tr>
<tr>
<td>EventBus Namespace</td>
<td>2197</td>
</tr>
<tr>
<td>ChangeEventHeader Class</td>
<td>2197</td>
</tr>
<tr>
<td>EventPublishFailureCallback Interface</td>
<td>2203</td>
</tr>
<tr>
<td>EventPublishSuccessCallback Interface</td>
<td>2204</td>
</tr>
<tr>
<td>FailureResult Interface</td>
<td>2205</td>
</tr>
<tr>
<td>SuccessResult Interface</td>
<td>2206</td>
</tr>
<tr>
<td>TestBroker Class</td>
<td>2206</td>
</tr>
<tr>
<td>TriggerContext Class</td>
<td>2208</td>
</tr>
<tr>
<td>ExternalService Namespace</td>
<td>2211</td>
</tr>
<tr>
<td>Flow Namespace</td>
<td>2211</td>
</tr>
<tr>
<td>Interview Class</td>
<td>2211</td>
</tr>
<tr>
<td>Functions Namespace</td>
<td>2216</td>
</tr>
<tr>
<td>Function Class</td>
<td>2216</td>
</tr>
<tr>
<td>FunctionCallback Interface</td>
<td>2221</td>
</tr>
<tr>
<td>FunctionErrorType Enum</td>
<td>2222</td>
</tr>
<tr>
<td>FunctionInvocation Interface</td>
<td>2222</td>
</tr>
<tr>
<td>FunctionInvocationError Interface</td>
<td>2225</td>
</tr>
<tr>
<td>FunctionInvocationStatus Enum</td>
<td>2226</td>
</tr>
<tr>
<td>FunctionInvokeMock Interface</td>
<td>2227</td>
</tr>
<tr>
<td>MockFunctionInvocationFactory Class</td>
<td>2230</td>
</tr>
<tr>
<td>Invocable Namespace</td>
<td>2232</td>
</tr>
<tr>
<td>Action Class</td>
<td>2232</td>
</tr>
<tr>
<td>Action.Error Class</td>
<td>2237</td>
</tr>
<tr>
<td>Action.Result Class</td>
<td>2239</td>
</tr>
<tr>
<td>KbManagement Namespace</td>
<td>2241</td>
</tr>
<tr>
<td>PublishingService Class</td>
<td>2241</td>
</tr>
<tr>
<td>LxScheduler Namespace</td>
<td>2252</td>
</tr>
<tr>
<td>GetAppointmentCandidatesInput Class</td>
<td>2253</td>
</tr>
<tr>
<td>GetAppointmentCandidatesInputBuilder Class</td>
<td>2256</td>
</tr>
<tr>
<td>GetAppointmentSlotsInput Class</td>
<td>2263</td>
</tr>
<tr>
<td>GetAppointmentSlotsInputBuilder Class</td>
<td>2265</td>
</tr>
<tr>
<td>SchedulerResources Class</td>
<td>2272</td>
</tr>
<tr>
<td>SkillRequirement Class</td>
<td>2277</td>
</tr>
<tr>
<td>SkillRequirementBuilder Class</td>
<td>2277</td>
</tr>
<tr>
<td>Class/Interface/Enum</td>
<td>Page</td>
</tr>
<tr>
<td>-----------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>WorkType Class</td>
<td>2279</td>
</tr>
<tr>
<td>WorkTypeBuilder Class</td>
<td>2279</td>
</tr>
<tr>
<td>ServiceResourceScheduleHandler Interface</td>
<td>2283</td>
</tr>
<tr>
<td>ServiceAppointmentRequestInfo Class</td>
<td>2286</td>
</tr>
<tr>
<td>ServiceResourceInfo Class</td>
<td>2290</td>
</tr>
<tr>
<td>ServiceResourceSchedule Class</td>
<td>2293</td>
</tr>
<tr>
<td>UnavailableTimeslot Class</td>
<td>2295</td>
</tr>
<tr>
<td>Messaging Namespace</td>
<td>2297</td>
</tr>
<tr>
<td>Attachmen RetrievalOption Enum</td>
<td>2298</td>
</tr>
<tr>
<td>Email Class (Base Email Methods)</td>
<td>2298</td>
</tr>
<tr>
<td>EmailFileAttachment Class</td>
<td>2302</td>
</tr>
<tr>
<td>InboundEmail Class</td>
<td>2304</td>
</tr>
<tr>
<td>InboundEmail.AuthenticationResult Class</td>
<td>2310</td>
</tr>
<tr>
<td>InboundEmail.AuthenticationResultField Class</td>
<td>2312</td>
</tr>
<tr>
<td>InboundEmail.BinaryAttachment Class</td>
<td>2313</td>
</tr>
<tr>
<td>InboundEmail.TextAttachment Class</td>
<td>2315</td>
</tr>
<tr>
<td>InboundEmailResult Class</td>
<td>2318</td>
</tr>
<tr>
<td>InboundEnvelope Class</td>
<td>2319</td>
</tr>
<tr>
<td>MassEmailMessage Class</td>
<td>2320</td>
</tr>
<tr>
<td>InboundEmail.Header Class</td>
<td>2322</td>
</tr>
<tr>
<td>PushNotification Class</td>
<td>2323</td>
</tr>
<tr>
<td>PushNotificationPayload Class</td>
<td>2327</td>
</tr>
<tr>
<td>CustomNotification Class</td>
<td>2329</td>
</tr>
<tr>
<td>RenderEmailTemplateBodyResult Class</td>
<td>2336</td>
</tr>
<tr>
<td>RenderEmailTemplateError Class</td>
<td>2338</td>
</tr>
<tr>
<td>SendEmailError Class</td>
<td>2339</td>
</tr>
<tr>
<td>SendEmailResult Class</td>
<td>2341</td>
</tr>
<tr>
<td>SingleEmailMessage Methods</td>
<td>2342</td>
</tr>
<tr>
<td>Metadata Namespace</td>
<td>2354</td>
</tr>
<tr>
<td>AnalyticsCloudComponentLayoutItem Class</td>
<td>2357</td>
</tr>
<tr>
<td>ConsoleComponent Class</td>
<td>2361</td>
</tr>
<tr>
<td>Container Class</td>
<td>2363</td>
</tr>
<tr>
<td>CustomConsoleComponents Class</td>
<td>2366</td>
</tr>
<tr>
<td>CustomMetadata Class</td>
<td>2367</td>
</tr>
<tr>
<td>CustomMetadataValue Class</td>
<td>2370</td>
</tr>
<tr>
<td>DeployCallback Interface</td>
<td>2372</td>
</tr>
<tr>
<td>DeployCallbackContext Class</td>
<td>2373</td>
</tr>
<tr>
<td>DeployContainer Class</td>
<td>2375</td>
</tr>
<tr>
<td>DeployDetails Class</td>
<td>2377</td>
</tr>
<tr>
<td>DeployMessage Class</td>
<td>2379</td>
</tr>
<tr>
<td>DeployProblemType Enum</td>
<td>2384</td>
</tr>
<tr>
<td>DeployResult Class</td>
<td>2384</td>
</tr>
<tr>
<td>DeployStatus Enum</td>
<td>2392</td>
</tr>
<tr>
<td>FeedItemTypeEnum</td>
<td>2393</td>
</tr>
</tbody>
</table>
Contents

FeedLayout Class .................................................. 2394
FeedLayoutComponent Class ...................................... 2398
FeedLayoutComponentType Enum .................................. 2400
FeedLayoutFilter Class ........................................... 2400
FeedLayoutFilterPosition Enum ................................. 2402
FeedLayoutFilterType Enum ..................................... 2403
Layout Class ........................................................... 2403
LayoutColumn Class ................................................ 2410
LayoutHeader Enum .................................................. 2412
LayoutItem Class .................................................... 2412
LayoutSection Class ................................................ 2417
LayoutSectionStyle Enum ......................................... 2420
Metadata Class ...................................................... 2420
MetadataType Enum ................................................ 2422
MetadataValue Class ................................................ 2422
MiniLayout Class ..................................................... 2423
Operations Class .................................................... 2424
PlatformActionList Class .......................................... 2428
PlatformActionListContextEnum Enum .......................... 2430
PlatformActionListItem Class ..................................... 2431
PlatformActionTypeEnum Enum .................................. 2433
PrimaryTabComponents Class ................................... 2433
QuickActionList Class .............................................. 2435
QuickActionListItem Class ....................................... 2436
RelatedContent Class .............................................. 2437
RelatedContentItem Class ....................................... 2438
RelatedList Class ................................................... 2440
RelatedListItem Class ............................................. 2441
ReportChartComponentLayoutItem Class ...................... 2444
ReportChartComponentSize Enum ................................ 2447
SidebarComponent Class .......................................... 2448
SortOrder Enum ....................................................... 2452
StatusCode Enum .................................................... 2452
SubtabComponents Class ......................................... 2453
SummaryLayoutStyleEnum Enum ................................ 2454
SummaryLayout Class .............................................. 2455
SummaryLayoutItem Class ........................................ 2458
UiBehavior Enum ..................................................... 2460
Pref_center Namespace ........................................... 2460
LoadFormData Class ................................................ 2461
LoadParameters Class .............................................. 2467
PreferenceCenterApexHandler Interface ....................... 2468
SubmitFormData Class .............................................. 2470
SubmitParameters Class .......................................... 2473
<table>
<thead>
<tr>
<th>Namespace</th>
<th>Class/Interface</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>TokenType Enum</td>
<td></td>
<td>2474</td>
</tr>
<tr>
<td>TokenUtility Class</td>
<td></td>
<td>2475</td>
</tr>
<tr>
<td>ValidationResult Class</td>
<td></td>
<td>2477</td>
</tr>
<tr>
<td>Process Namespace</td>
<td>Plugin Interface</td>
<td>2478</td>
</tr>
<tr>
<td></td>
<td>PluginDescribeResult Class</td>
<td>2480</td>
</tr>
<tr>
<td></td>
<td>PluginDescribeResult.InputParameter Class</td>
<td>2483</td>
</tr>
<tr>
<td></td>
<td>PluginDescribeResult.OutputParameter Class</td>
<td>2486</td>
</tr>
<tr>
<td></td>
<td>PluginRequest Class</td>
<td>2489</td>
</tr>
<tr>
<td></td>
<td>PluginResult Class</td>
<td>2489</td>
</tr>
<tr>
<td>QuickAction Namespace</td>
<td>DescribeAvailableQuickActionResult Class</td>
<td>2491</td>
</tr>
<tr>
<td></td>
<td>DescribeLayoutComponent Class</td>
<td>2493</td>
</tr>
<tr>
<td></td>
<td>DescribeLayoutItem Class</td>
<td>2494</td>
</tr>
<tr>
<td></td>
<td>DescribeLayoutRow Class</td>
<td>2496</td>
</tr>
<tr>
<td></td>
<td>DescribeLayoutSection Class</td>
<td>2497</td>
</tr>
<tr>
<td></td>
<td>DescribeQuickActionDefaultValue Class</td>
<td>2501</td>
</tr>
<tr>
<td></td>
<td>DescribeQuickActionResult Class</td>
<td>2502</td>
</tr>
<tr>
<td></td>
<td>QuickActionDefaults Class</td>
<td>2518</td>
</tr>
<tr>
<td></td>
<td>QuickActionDefaultsHandler Interface</td>
<td>2520</td>
</tr>
<tr>
<td></td>
<td>QuickActionRequest Class</td>
<td>2525</td>
</tr>
<tr>
<td></td>
<td>QuickActionResult Class</td>
<td>2528</td>
</tr>
<tr>
<td></td>
<td>SendEmailQuickActionDefaults Class</td>
<td>2530</td>
</tr>
<tr>
<td>Reports Namespace</td>
<td>AggregateColumn Class</td>
<td>2536</td>
</tr>
<tr>
<td></td>
<td>BucketField Class</td>
<td>2537</td>
</tr>
<tr>
<td></td>
<td>BucketFieldValue Class</td>
<td>2544</td>
</tr>
<tr>
<td></td>
<td>BucketTypeEnum</td>
<td>2548</td>
</tr>
<tr>
<td></td>
<td>ColumnDataType Enum</td>
<td>2548</td>
</tr>
<tr>
<td></td>
<td>ColumnSortOrder Enum</td>
<td>2549</td>
</tr>
<tr>
<td></td>
<td>CrossFilter Class</td>
<td>2550</td>
</tr>
<tr>
<td></td>
<td>CsfGroupTypeEnum</td>
<td>2555</td>
</tr>
<tr>
<td></td>
<td>DateGranularityEnum</td>
<td>2555</td>
</tr>
<tr>
<td></td>
<td>DetailColumn Class</td>
<td>2556</td>
</tr>
<tr>
<td></td>
<td>Dimension Class</td>
<td>2557</td>
</tr>
<tr>
<td></td>
<td>EvaluatedCondition Class</td>
<td>2558</td>
</tr>
<tr>
<td></td>
<td>EvaluatedConditionOperatorEnum</td>
<td>2561</td>
</tr>
<tr>
<td></td>
<td>FilterOperator Class</td>
<td>2562</td>
</tr>
<tr>
<td></td>
<td>FilterValue Class</td>
<td>2563</td>
</tr>
<tr>
<td></td>
<td>FormulaTypeEnum</td>
<td>2564</td>
</tr>
<tr>
<td></td>
<td>GroupingColumn Class</td>
<td>2564</td>
</tr>
<tr>
<td></td>
<td>GroupingInfo Class</td>
<td>2566</td>
</tr>
<tr>
<td></td>
<td>GroupingValue Class</td>
<td>2567</td>
</tr>
<tr>
<td></td>
<td>NotificationAction Interface</td>
<td>2569</td>
</tr>
<tr>
<td>Class Name</td>
<td>Page</td>
<td></td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td>NotificationActionContext Class</td>
<td>2571</td>
<td></td>
</tr>
<tr>
<td>ReportCsf Class</td>
<td>2572</td>
<td></td>
</tr>
<tr>
<td>ReportCurrency Class</td>
<td>2581</td>
<td></td>
</tr>
<tr>
<td>ReportDataCell Class</td>
<td>2582</td>
<td></td>
</tr>
<tr>
<td>ReportDescribeResult Class</td>
<td>2583</td>
<td></td>
</tr>
<tr>
<td>ReportDetailRow Class</td>
<td>2584</td>
<td></td>
</tr>
<tr>
<td>ReportDivisionInfo Class</td>
<td>2585</td>
<td></td>
</tr>
<tr>
<td>ReportExtendedMetadata Class</td>
<td>2586</td>
<td></td>
</tr>
<tr>
<td>ReportFact Class</td>
<td>2587</td>
<td></td>
</tr>
<tr>
<td>ReportFactWithDetails Class</td>
<td>2588</td>
<td></td>
</tr>
<tr>
<td>ReportFactWithSummaries Class</td>
<td>2589</td>
<td></td>
</tr>
<tr>
<td>ReportFilter Class</td>
<td>2591</td>
<td></td>
</tr>
<tr>
<td>ReportFormat Enum</td>
<td>2595</td>
<td></td>
</tr>
<tr>
<td>ReportFilterType Enum</td>
<td>2596</td>
<td></td>
</tr>
<tr>
<td>ReportInstance Class</td>
<td>2596</td>
<td></td>
</tr>
<tr>
<td>ReportManager Class</td>
<td>2599</td>
<td></td>
</tr>
<tr>
<td>ReportMetadata Class</td>
<td>2604</td>
<td></td>
</tr>
<tr>
<td>ReportResults Class</td>
<td>2624</td>
<td></td>
</tr>
<tr>
<td>ReportScopeInfo Class</td>
<td>2626</td>
<td></td>
</tr>
<tr>
<td>ReportScopeValue Class</td>
<td>2627</td>
<td></td>
</tr>
<tr>
<td>ReportType Class</td>
<td>2628</td>
<td></td>
</tr>
<tr>
<td>ReportTypeColumn Class</td>
<td>2629</td>
<td></td>
</tr>
<tr>
<td>ReportTypeColumnCategory Class</td>
<td>2631</td>
<td></td>
</tr>
<tr>
<td>ReportTypeMetadata Class</td>
<td>2633</td>
<td></td>
</tr>
<tr>
<td>SortColumn Class</td>
<td>2635</td>
<td></td>
</tr>
<tr>
<td>StandardDateFilter Class</td>
<td>2636</td>
<td></td>
</tr>
<tr>
<td>StandardDateFilterDuration Class</td>
<td>2639</td>
<td></td>
</tr>
<tr>
<td>StandardDateFilterDurationGroup Class</td>
<td>2641</td>
<td></td>
</tr>
<tr>
<td>StandardFilter Class</td>
<td>2642</td>
<td></td>
</tr>
<tr>
<td>StandardFilterInfo Class</td>
<td>2644</td>
<td></td>
</tr>
<tr>
<td>StandardFilterInfoPicklist Class</td>
<td>2645</td>
<td></td>
</tr>
<tr>
<td>StandardFilterType Enum</td>
<td>2647</td>
<td></td>
</tr>
<tr>
<td>SummaryValue Class</td>
<td>2647</td>
<td></td>
</tr>
<tr>
<td>ThresholdInformation Class</td>
<td>2648</td>
<td></td>
</tr>
<tr>
<td>TopRows Class</td>
<td>2649</td>
<td></td>
</tr>
<tr>
<td>Reports Exceptions</td>
<td>2652</td>
<td></td>
</tr>
<tr>
<td>RichMessaging Namespace</td>
<td>2653</td>
<td></td>
</tr>
<tr>
<td>AbstractTiming Class</td>
<td>2654</td>
<td></td>
</tr>
<tr>
<td>AddressableContact Class</td>
<td>2654</td>
<td></td>
</tr>
<tr>
<td>AuthRequestHandler Interface</td>
<td>2658</td>
<td></td>
</tr>
<tr>
<td>AuthRequestResponse Class</td>
<td>2660</td>
<td></td>
</tr>
<tr>
<td>AuthRequestResult Class</td>
<td>2662</td>
<td></td>
</tr>
<tr>
<td>AuthRequestResultStatus Enum</td>
<td>2664</td>
<td></td>
</tr>
<tr>
<td>DeferredTiming Class</td>
<td>2664</td>
<td></td>
</tr>
<tr>
<td>Namespace</td>
<td>Page</td>
<td></td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>-------</td>
<td></td>
</tr>
<tr>
<td>Sfc Namespace</td>
<td>2785</td>
<td></td>
</tr>
<tr>
<td>ContentDownloadContext Enum</td>
<td>2786</td>
<td></td>
</tr>
<tr>
<td>ContentDownloadHandler Class</td>
<td>2786</td>
<td></td>
</tr>
<tr>
<td>ContentDownloadHandlerFactory Interface</td>
<td>2788</td>
<td></td>
</tr>
<tr>
<td>Sfdc Checkout Namespace</td>
<td>2789</td>
<td></td>
</tr>
<tr>
<td>AsyncCartProcessor Interface</td>
<td>2790</td>
<td></td>
</tr>
<tr>
<td>B2BCheckoutController Class</td>
<td>2791</td>
<td></td>
</tr>
<tr>
<td>IntegrationInfo Class</td>
<td>2792</td>
<td></td>
</tr>
<tr>
<td>IntegrationStatus Class</td>
<td>2793</td>
<td></td>
</tr>
<tr>
<td>IntegrationStatus.Status Enum</td>
<td>2794</td>
<td></td>
</tr>
<tr>
<td>sfdc_surveys Namespace</td>
<td>2794</td>
<td></td>
</tr>
<tr>
<td>SurveyInvitationLinkShortener Interface</td>
<td>2795</td>
<td></td>
</tr>
<tr>
<td>Example Implementation to Associate SurveySubjects with SurveyInvitation and SurveyResponses</td>
<td>2796</td>
<td></td>
</tr>
<tr>
<td>Site Namespace</td>
<td>2798</td>
<td></td>
</tr>
<tr>
<td>UrlRewriter Interface</td>
<td>2798</td>
<td></td>
</tr>
<tr>
<td>Site Exceptions</td>
<td>2800</td>
<td></td>
</tr>
<tr>
<td>Support Namespace</td>
<td>2800</td>
<td></td>
</tr>
<tr>
<td>EmailTemplateSelector Interface</td>
<td>2800</td>
<td></td>
</tr>
<tr>
<td>MilestoneTriggerTimeCalculator Interface</td>
<td>2802</td>
<td></td>
</tr>
<tr>
<td>System Namespace</td>
<td>2804</td>
<td></td>
</tr>
<tr>
<td>AccessLevel Class</td>
<td>2811</td>
<td></td>
</tr>
<tr>
<td>AccessType Enum</td>
<td>2814</td>
<td></td>
</tr>
<tr>
<td>Address Class</td>
<td>2815</td>
<td></td>
</tr>
<tr>
<td>Answers Class</td>
<td>2820</td>
<td></td>
</tr>
<tr>
<td>ApexPages Class</td>
<td>2822</td>
<td></td>
</tr>
<tr>
<td>Approval Class</td>
<td>2825</td>
<td></td>
</tr>
<tr>
<td>Assert Class</td>
<td>2836</td>
<td></td>
</tr>
<tr>
<td>AsyncInfo Class</td>
<td>2849</td>
<td></td>
</tr>
<tr>
<td>AsyncOptions Class</td>
<td>2851</td>
<td></td>
</tr>
<tr>
<td>Blob Class</td>
<td>2852</td>
<td></td>
</tr>
<tr>
<td>Boolean Class</td>
<td>2854</td>
<td></td>
</tr>
<tr>
<td>BusinessHours Class</td>
<td>2856</td>
<td></td>
</tr>
<tr>
<td>Callable Interface</td>
<td>2860</td>
<td></td>
</tr>
<tr>
<td>Cases Class</td>
<td>2862</td>
<td></td>
</tr>
<tr>
<td>Collator Class</td>
<td>2866</td>
<td></td>
</tr>
<tr>
<td>Comparable Interface</td>
<td>2868</td>
<td></td>
</tr>
<tr>
<td>Comparator Interface</td>
<td>2870</td>
<td></td>
</tr>
<tr>
<td>Continuation Class</td>
<td>2873</td>
<td></td>
</tr>
<tr>
<td>Cookie Class</td>
<td>2877</td>
<td></td>
</tr>
<tr>
<td>Crypto Class</td>
<td>2883</td>
<td></td>
</tr>
<tr>
<td>Custom Metadata Type Methods</td>
<td>2899</td>
<td></td>
</tr>
<tr>
<td>Custom Settings Methods</td>
<td>2902</td>
<td></td>
</tr>
<tr>
<td>Database Class</td>
<td>2913</td>
<td></td>
</tr>
<tr>
<td>Class/Enum</td>
<td>Page</td>
<td></td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td>Date Class</td>
<td>3005</td>
<td></td>
</tr>
<tr>
<td>Datetime Class</td>
<td>3015</td>
<td></td>
</tr>
<tr>
<td>Decimal Class</td>
<td>3039</td>
<td></td>
</tr>
<tr>
<td>Domain Class</td>
<td>3052</td>
<td></td>
</tr>
<tr>
<td>DomainCreator Class</td>
<td>3055</td>
<td></td>
</tr>
<tr>
<td>DomainParser Class</td>
<td>3059</td>
<td></td>
</tr>
<tr>
<td>DomainType Enum</td>
<td>3061</td>
<td></td>
</tr>
<tr>
<td>Double Class</td>
<td>3062</td>
<td></td>
</tr>
<tr>
<td>EmailMessages Class</td>
<td>3065</td>
<td></td>
</tr>
<tr>
<td>EncodingUtil Class</td>
<td>3069</td>
<td></td>
</tr>
<tr>
<td>Enum Methods</td>
<td>3073</td>
<td></td>
</tr>
<tr>
<td>EventBus Class</td>
<td>3073</td>
<td></td>
</tr>
<tr>
<td>Exception Class and Built-In Exceptions</td>
<td>3076</td>
<td></td>
</tr>
<tr>
<td>FlexQueue Class</td>
<td>3081</td>
<td></td>
</tr>
<tr>
<td>FeatureManagement Class</td>
<td>3084</td>
<td></td>
</tr>
<tr>
<td>Formula Class</td>
<td>3089</td>
<td></td>
</tr>
<tr>
<td>FormulaRecalcFieldError Class</td>
<td>3090</td>
<td></td>
</tr>
<tr>
<td>FormulaRecalcResult Class</td>
<td>3091</td>
<td></td>
</tr>
<tr>
<td>Http Class</td>
<td>3093</td>
<td></td>
</tr>
<tr>
<td>HttpCalloutMock Interface</td>
<td>3094</td>
<td></td>
</tr>
<tr>
<td>HttpRequest Class</td>
<td>3095</td>
<td></td>
</tr>
<tr>
<td>HttpResponse Class</td>
<td>3104</td>
<td></td>
</tr>
<tr>
<td>Id Class</td>
<td>3111</td>
<td></td>
</tr>
<tr>
<td>Ideas Class</td>
<td>3117</td>
<td></td>
</tr>
<tr>
<td>InstallHandler Interface</td>
<td>3122</td>
<td></td>
</tr>
<tr>
<td>Integer Class</td>
<td>3125</td>
<td></td>
</tr>
<tr>
<td>JSON Class</td>
<td>3128</td>
<td></td>
</tr>
<tr>
<td>JSONGenerator Class</td>
<td>3134</td>
<td></td>
</tr>
<tr>
<td>JSONParser Class</td>
<td>3148</td>
<td></td>
</tr>
<tr>
<td>JSONToken Enum</td>
<td>3160</td>
<td></td>
</tr>
<tr>
<td>Label Class</td>
<td>3161</td>
<td></td>
</tr>
<tr>
<td>Limits Class</td>
<td>3164</td>
<td></td>
</tr>
<tr>
<td>List Class</td>
<td>3177</td>
<td></td>
</tr>
<tr>
<td>Location Class</td>
<td>3192</td>
<td></td>
</tr>
<tr>
<td>LoggingLevel Enum</td>
<td>3195</td>
<td></td>
</tr>
<tr>
<td>Long Class</td>
<td>3196</td>
<td></td>
</tr>
<tr>
<td>Map Class</td>
<td>3198</td>
<td></td>
</tr>
<tr>
<td>Matcher Class</td>
<td>3210</td>
<td></td>
</tr>
<tr>
<td>Math Class</td>
<td>3222</td>
<td></td>
</tr>
<tr>
<td>Messaging Class</td>
<td>3248</td>
<td></td>
</tr>
<tr>
<td>MultiStaticResourceCalloutMock Class</td>
<td>3256</td>
<td></td>
</tr>
<tr>
<td>Network Class</td>
<td>3259</td>
<td></td>
</tr>
<tr>
<td>OrgLimit Class</td>
<td>3264</td>
<td></td>
</tr>
<tr>
<td>OrgLimits Class</td>
<td>3267</td>
<td></td>
</tr>
<tr>
<td>Class/Interface</td>
<td>Page</td>
<td></td>
</tr>
<tr>
<td>-----------------</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td>PageReference Class</td>
<td>3268</td>
<td></td>
</tr>
<tr>
<td>Packaging Class</td>
<td>3280</td>
<td></td>
</tr>
<tr>
<td>Pattern Class</td>
<td>3281</td>
<td></td>
</tr>
<tr>
<td>Queueable Interface</td>
<td>3285</td>
<td></td>
</tr>
<tr>
<td>QueueableContext Interface</td>
<td>3287</td>
<td></td>
</tr>
<tr>
<td>QueueableDuplicateSignature Class</td>
<td>3288</td>
<td></td>
</tr>
<tr>
<td>QuickAction Class</td>
<td>3292</td>
<td></td>
</tr>
<tr>
<td>Quiddity Enum</td>
<td>3297</td>
<td></td>
</tr>
<tr>
<td>RemoteObjectController</td>
<td>3298</td>
<td></td>
</tr>
<tr>
<td>Request Class</td>
<td>3301</td>
<td></td>
</tr>
<tr>
<td>ResetPasswordResult Class</td>
<td>3303</td>
<td></td>
</tr>
<tr>
<td>RestContext Class</td>
<td>3304</td>
<td></td>
</tr>
<tr>
<td>RestRequest Class</td>
<td>3305</td>
<td></td>
</tr>
<tr>
<td>RestResponse Class</td>
<td>3310</td>
<td></td>
</tr>
<tr>
<td>SandboxPostCopy Interface</td>
<td>3314</td>
<td></td>
</tr>
<tr>
<td>Schedulable Interface</td>
<td>3317</td>
<td></td>
</tr>
<tr>
<td>SchedulableContext Interface</td>
<td>3318</td>
<td></td>
</tr>
<tr>
<td>Schema Class</td>
<td>3318</td>
<td></td>
</tr>
<tr>
<td>Search Class</td>
<td>3323</td>
<td></td>
</tr>
<tr>
<td>Security Class</td>
<td>3329</td>
<td></td>
</tr>
<tr>
<td>SelectOption Class</td>
<td>3333</td>
<td></td>
</tr>
<tr>
<td>Set Class</td>
<td>3339</td>
<td></td>
</tr>
<tr>
<td>Site Class</td>
<td>3350</td>
<td></td>
</tr>
<tr>
<td>SObject Class</td>
<td>3371</td>
<td></td>
</tr>
<tr>
<td>SObjectAccessDecision Class</td>
<td>3397</td>
<td></td>
</tr>
<tr>
<td>StaticResourceCalloutMock Class</td>
<td>3399</td>
<td></td>
</tr>
<tr>
<td>String Class</td>
<td>3402</td>
<td></td>
</tr>
<tr>
<td>StubProvider Interface</td>
<td>3477</td>
<td></td>
</tr>
<tr>
<td>System Class</td>
<td>3479</td>
<td></td>
</tr>
<tr>
<td>Test Class</td>
<td>3503</td>
<td></td>
</tr>
<tr>
<td>Time Class</td>
<td>3521</td>
<td></td>
</tr>
<tr>
<td>TimeZone Class</td>
<td>3526</td>
<td></td>
</tr>
<tr>
<td>Trigger Class</td>
<td>3529</td>
<td></td>
</tr>
<tr>
<td>TriggerOperation Enum</td>
<td>3532</td>
<td></td>
</tr>
<tr>
<td>Type Class</td>
<td>3532</td>
<td></td>
</tr>
<tr>
<td>UninstallHandler Interface</td>
<td>3540</td>
<td></td>
</tr>
<tr>
<td>URL Class</td>
<td>3542</td>
<td></td>
</tr>
<tr>
<td>UserInfo Class</td>
<td>3552</td>
<td></td>
</tr>
<tr>
<td>UserManagement Class</td>
<td>3561</td>
<td></td>
</tr>
<tr>
<td>Version Class</td>
<td>3580</td>
<td></td>
</tr>
<tr>
<td>WebServiceCallout Class</td>
<td>3584</td>
<td></td>
</tr>
<tr>
<td>WebServiceMock Interface</td>
<td>3585</td>
<td></td>
</tr>
<tr>
<td>XmlStreamReader Class</td>
<td>3587</td>
<td></td>
</tr>
<tr>
<td>XmlStreamWriter Class</td>
<td>3601</td>
<td></td>
</tr>
<tr>
<td>Namespace</td>
<td>Page</td>
<td></td>
</tr>
<tr>
<td>-----------</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td>TerritoryMgmt Namespace</td>
<td>3608</td>
<td></td>
</tr>
<tr>
<td>OpportunityTerritory2AssignmentFilter Global Interface</td>
<td>3608</td>
<td></td>
</tr>
<tr>
<td>TxnSecurity Namespace</td>
<td>3612</td>
<td></td>
</tr>
<tr>
<td>Event Class</td>
<td>3612</td>
<td></td>
</tr>
<tr>
<td>EventCondition Interface</td>
<td>3616</td>
<td></td>
</tr>
<tr>
<td>AsyncCondition Interface</td>
<td>3618</td>
<td></td>
</tr>
<tr>
<td>PolicyCondition Interface</td>
<td>3619</td>
<td></td>
</tr>
<tr>
<td>UserProvisioning Namespace</td>
<td>3621</td>
<td></td>
</tr>
<tr>
<td>ConnectorTestUtil Class</td>
<td>3621</td>
<td></td>
</tr>
<tr>
<td>UserProvisioningLog Class</td>
<td>3623</td>
<td></td>
</tr>
<tr>
<td>UserProvisioningPlugin Class</td>
<td>3625</td>
<td></td>
</tr>
<tr>
<td>VisualEditor Namespace</td>
<td>3629</td>
<td></td>
</tr>
<tr>
<td>DataRow Class</td>
<td>3630</td>
<td></td>
</tr>
<tr>
<td>DesignTimePageContext Class</td>
<td>3633</td>
<td></td>
</tr>
<tr>
<td>DynamicPickList Class</td>
<td>3635</td>
<td></td>
</tr>
<tr>
<td>DynamicPickListRows Class</td>
<td>3638</td>
<td></td>
</tr>
<tr>
<td>Wave Namespace</td>
<td>3643</td>
<td></td>
</tr>
<tr>
<td>QueryBuilder Class</td>
<td>3643</td>
<td></td>
</tr>
<tr>
<td>QueryNode Class</td>
<td>3647</td>
<td></td>
</tr>
<tr>
<td>ProjectionNode Class</td>
<td>3651</td>
<td></td>
</tr>
<tr>
<td>Templates Class</td>
<td>3654</td>
<td></td>
</tr>
<tr>
<td>TemplatesSearchOptions Class</td>
<td>3657</td>
<td></td>
</tr>
<tr>
<td>Appendices</td>
<td>3659</td>
<td></td>
</tr>
<tr>
<td>Shipping Invoice Example</td>
<td>3659</td>
<td></td>
</tr>
<tr>
<td>Reserved Keywords</td>
<td>3671</td>
<td></td>
</tr>
<tr>
<td>Documentation Typographical Conventions</td>
<td>3672</td>
<td></td>
</tr>
<tr>
<td>Glossary</td>
<td>3673</td>
<td></td>
</tr>
</tbody>
</table>
Apex is a strongly typed, object-oriented programming language that allows developers to execute flow and transaction control statements on the Salesforce Platform server, in conjunction with calls to the API. This reference guide includes built-in Apex classes, interfaces, enum, and exceptions, grouped by namespace. It also includes Apex DML statements to insert, update, merge, delete, and restore data in Salesforce.

For information on the Apex development process, see Apex Developer Guide.

Note: In API version 51.0 and earlier, Apex Reference information was included in the Apex Developer Guide in the Apex Language section.

IN THIS SECTION:

- Apex Release Notes
  Use the Salesforce Release Notes to learn about the most recent updates and changes to Apex.
- Apex DML Operations
  You can perform DML operations using the Apex DML statements or the methods of the Database class. For lead conversion, use the convertLead method of the Database class. There is no DML counterpart for it.
- ApexPages Namespace
  The ApexPages namespace provides classes used in Visualforce controllers.
- AppLauncher Namespace
  The AppLauncher namespace provides methods for managing the appearance of apps in the App Launcher, including their visibility and sort order.
- Approval Namespace
  The Approval namespace provides classes and methods for approval processes.
- Auth Namespace
  The Auth namespace provides an interface and classes for single sign-on into Salesforce and session security management.
- Cache Namespace
  The Cache namespace contains methods for managing the platform cache.
- Canvas Namespace
  The Canvas namespace provides an interface and classes for canvas apps in Salesforce.
- ChatterAnswers Namespace
  The ChatterAnswers namespace provides an interface for creating Account records.
- CommercePayments Namespace
  Use the CommercePayments namespace to provide a safe and customizable platform for managing customer payments and refunds.
- ConnectApi Namespace
  The ConnectApi namespace (also called Connect in Apex) provides classes for accessing the same data available in Connect REST API. Use Connect in Apex to create custom experiences in Salesforce.
- Database Namespace
  The Database namespace provides classes used with DML operations.
Datacloud Namespace
The Datacloud namespace provides classes and methods for retrieving information about duplicate rules. Duplicate rules let you control whether and when users can save duplicate records within Salesforce.

DataSource Namespace
The DataSource namespace provides the classes for the Apex Connector Framework. Use the Apex Connector Framework to develop a custom adapter for Salesforce Connect. Then connect your Salesforce organization to any data anywhere via the Salesforce Connect custom adapter.

DataWeave Namespace
The DataWeave namespace provides classes and methods to support the invocation of DataWeave scripts from Apex.

Dom Namespace
The Dom namespace provides classes and methods for parsing and creating XML content.

EventBus Namespace
The EventBus namespace provides classes and methods for platform events and Change Data Capture events.

ExternalService Namespace
The ExternalService namespace provides dynamically generated Apex service interfaces and Apex classes for complex object data types.

Flow Namespace
The Flow namespace provides a class for advanced Visualforce controller access to flows.

Functions Namespace
The Functions namespace provides classes and methods used to invoke and manage Salesforce Functions.

Invocable Namespace
The Invocable namespace provides classes for calling invocable actions from Apex.

KbManagement Namespace
The KbManagement namespace provides a class for managing knowledge articles.

LxScheduler Namespace
The LxScheduler namespace provides an interface and classes for integrating Salesforce Scheduler with external calendars.

Messaging Namespace
The Messaging namespace provides classes and methods for Salesforce outbound and inbound email functionality.

Metadata Namespace
The Metadata namespace provides classes and methods for working with custom metadata in Salesforce

Pref_center Namespace
The Pref_center namespace provides an interface, classes, and methods to create and retrieve data in forms in Preference Manager. Preference Manager, previously called Preference Center, is a feature within the Privacy Center app.

Process Namespace
The Process namespace provides an interface and classes for passing data between your organization and a flow.

QuickAction Namespace
The QuickAction namespace provides classes and methods for quick actions.

Reports Namespace
The Reports namespace provides classes for accessing the same data as is available in the Salesforce Reports and Dashboards REST API.
RichMessaging Namespace
Provides objects and methods for handling content in enhanced Messaging channels.

Schema Namespace
The Schema namespace provides classes and methods for schema metadata information.

Search Namespace
The Search namespace provides classes for getting search results and suggestion results.

Sfc Namespace
The Sfc namespace contains classes used in Salesforce Files.

Sfdc_Checkout Namespace
The Sfdc_Checkout namespace provides an interface and classes for B2B Commerce apps in Salesforce.

sfdc_surveys Namespace
The sfdc_surveys namespace provides an interface for shortening survey invitations.

Site Namespace
The Site namespace provides an interface for rewriting Sites URLs.

Support Namespace
The Support namespace provides an interface used for Case Feed.

System Namespace
The System namespace provides classes and methods for core Apex functionality.

TerritoryMgmt Namespace
The TerritoryMgmt namespace provides an interface used for territory management.

TxnSecurity Namespace
The TxnSecurity namespace provides an interface used for transaction security.

UserProvisioning Namespace
The UserProvisioning namespace provides methods for monitoring outbound user provisioning requests.

VisualEditor Namespace
The VisualEditor namespace provides classes and methods for interacting with the Lightning App Builder. The classes and methods in this namespace operate on Lightning components, which include Lightning web components and Aura components.

Wave Namespace
The classes in the Wave namespace are part of the CRM Analytics Analytics SDK, designed to facilitate querying CRM Analytics data from Apex code.

Appendices

Apex Release Notes

Use the Salesforce Release Notes to learn about the most recent updates and changes to Apex.

For Apex updates and changes that impact the Salesforce Platform, see the Apex Release Notes.

For new and changed Apex classes, methods, exceptions and interfaces, see Apex: New and Changed Items in the Salesforce Release Notes.
Apex DML Operations

You can perform DML operations using the Apex DML statements or the methods of the `Database` class. For lead conversion, use the `convertLead` method of the `Database` class. There is no DML counterpart for it.

SEE ALSO:

* Apex Developer Guide: Working with Data in Apex
  * Database Class

Apex DML Statements

Use Data Manipulation Language (DML) statements to insert, update, merge, delete, and restore data in Salesforce.

The following Apex DML statements are available:

### Insert Statement

The `insert` DML operation adds one or more sObjects, such as individual accounts or contacts, to your organization’s data. `insert` is analogous to the INSERT statement in SQL.

**Syntax**

```
insert sObject
insert sObject[]
```

**Example**

The following example inserts an account named ‘Acme’:

```apex
Account newAcct = new Account(name = 'Acme');
try {
    insert newAcct;
} catch (DmlException e) {
    // Process exception here
}
```

⚠️ Note: For more information on processing `DmlExceptions`, see [Bulk DML Exception Handling](#).

### Update Statement

The `update` DML operation modifies one or more existing sObject records, such as individual accounts or contacts, in your organization’s data. `update` is analogous to the UPDATE statement in SQL.

**Syntax**

```
update sObject
update sObject[]
```
Example
The following example updates the BillingCity field on a single account named ‘Acme’:

```java
Account a = new Account(Name='Acme2');
insert(a);

Account myAcct = [SELECT Id, Name, BillingCity FROM Account WHERE Id = :a.Id];
myAcct.BillingCity = 'San Francisco';

try {
    update myAcct;
} catch (DmlException e) {
    // Process exception here
}
```

Note: For more information on processing DmlExceptions, see Bulk DML Exception Handling.

Upsert Statement
The `upsert` DML operation creates new records and updates sObject records within a single statement, using a specified field to determine the presence of existing objects, or the ID field if no field is specified.

Syntax

```java
upsert sObject [opt_field]
upsert sObject[] [opt_field]
```

The `upsert` statement matches the sObjects with existing records by comparing values of one field. If you don’t specify a field when calling this statement, the `upsert` statement uses the sObject’s ID to match the sObject with existing records in Salesforce. Alternatively, you can specify a field to use for matching. For custom objects, specify a custom field marked as external ID. For standard objects, you can specify any field that has the `idLookup` attribute set to true. For example, the Email field of Contact or User has the `idLookup` attribute set. To check a field’s attribute, see the Object Reference for Salesforce.

Also, you can use foreign keys to upsert sObject records if they have been set as reference fields. For more information, see Field Types in the Object Reference for Salesforce.

The optional field parameter, `opt_field`, is a field token (of type `Schema.SObjectField`). For example, to specify the MyExternalId custom field, the statement is:

```java
upsert sObjectList Account.Fields.MyExternalId__c;
```

If the field used for matching doesn’t have the `Unique` attribute set, the context user must have the “View All” object-level permission for the target object or the “View All Data” permission so that `upsert` doesn’t accidentally insert a duplicate record.

Note: Custom field matching is case-insensitive only if the custom field has the `Unique` and `Treat "ABC" and "abc" as duplicate values (case insensitive)` attributes selected as part of the field definition. If so, “ABC123” is matched with “abc123.” For more information, see “Create Custom Fields” in the Salesforce online help.

How Upsert Chooses to Insert or Update
Upsert uses the sObject record’s primary key (the ID), an idLookup field, or an external ID field to determine whether it should create a record or update an existing one:
If the key isn’t matched, a new object record is created.
If the key is matched once, the existing object record is updated.
If the key is matched multiple times, an error is generated and the object record isn’t inserted or updated.

Example
This example performs an upsert of a list of accounts.

```java
List<Account> acctList = new List<Account>();
// Fill the accounts list with some accounts
try {
    upsert acctList;
} catch (DmlException e) {
}
```

This next example performs an upsert of a list of accounts using a foreign key for matching existing records, if any.

```java
List<Account> acctList = new List<Account>();
// Fill the accounts list with some accounts
try {
    // Upsert using an external ID field
    upsert acctList myExtIDField__c;
} catch (DmlException e) {
}
```

Delete Statement
The delete DML operation deletes one or more existing sObject records, such as individual accounts or contacts, from your organization’s data. delete is analogous to the delete() statement in the SOAP API.

Syntax
```
delete sObject
delete sObject[
```

Example
The following example deletes all accounts that are named 'DotCom':

```java
Account[] doomedAccts = [SELECT Id, Name FROM Account
WHERE Name = 'DotCom'];
try {
    delete doomedAccts;
} catch (DmlException e) {
    // Process exception here
}
```

Note: For more information on processing DmlExceptions, see Bulk DML Exception Handling.
Undelete Statement

The `undelete` DML operation restores one or more existing sObject records, such as individual accounts or contacts, from your organization's Recycle Bin. `undelete` is analogous to the UNDELETE statement in SQL.

Syntax

```
undelete sObject | ID
undelete sObject[] | ID[]
```

Example

The following example undeletes an account named 'Universal Containers'. The `ALL ROWS` keyword queries all rows for both top level and aggregate relationships, including deleted records and archived activities.

```
Account[] savedAccts = [SELECT Id, Name FROM Account WHERE Name = 'Universal Containers' ALL ROWS];
try {
    undelete savedAccts;
} catch (DmlException e) {
    // Process exception here
}
```

Note: For more information on processing `DmlExceptions`, see Bulk DML Exception Handling.

Merge Statement

The `merge` statement merges up to three records of the same sObject type into one of the records, deleting the others, and re-parenting any related records.

Note: This DML operation does not have a matching Database system method.

Syntax

```
merge sObject sObject
merge sObject sObject[]
merge sObject ID
merge sObject ID[]
```

The first parameter represents the master record into which the other records are to be merged. The second parameter represents the one or two other records that should be merged and then deleted. You can pass these other records into the `merge` statement as a single sObject record or ID, or as a list of two sObject records or IDs.

Example

The following example merges two accounts named 'Acme Inc.' and 'Acme' into a single record:

```
List<Account> ls = new List<Account>{new Account(name='Acme Inc.'),new Account(name='Acme')};
insert ls;
Account masterAcct = [SELECT Id, Name FROM Account WHERE Name = 'Acme Inc.' LIMIT 1];
```
Account mergeAcct = [SELECT Id, Name FROM Account WHERE Name = 'Acme' LIMIT 1];
try {
    merge masterAcct mergeAcct;
} catch (DmlException e) {
    // Process exception here
}

Note: For more information on processing DmlExceptions, see Bulk DML Exception Handling.

ApexPages Namespace

The ApexPages namespace provides classes used in Visualforce controllers. The following are the classes in the ApexPages namespace.

IN THIS SECTION:

- **Action Class**
  You can use `ApexPages.Action` to create an action method that you can use in a Visualforce custom controller or controller extension.

- **Component Class**
  Represents a dynamic Visualforce component in Apex.

- **IdeaStandardController Class**
  `IdeaStandardController` objects offer Ideas-specific functionality in addition to what is provided by the `StandardController`.

- **IdeaStandardSetController Class**
  `IdeaStandardSetController` objects offer Ideas-specific functionality in addition to what is provided by the `StandardSetController`.

- **KnowledgeArticleVersionStandardController Class**
  `KnowledgeArticleVersionStandardController` objects offer article-specific functionality in addition to what is provided by the `StandardController`.

- **Message Class**
  Contains validation errors that occur when the user saves the page that uses a standard controller.

- **StandardController Class**
  Use a `StandardController` when defining an extension for a standard controller.

- **StandardSetController Class**
  `StandardSetController` objects allow you to create list controllers similar to, or as extensions of, the pre-built Visualforce list controllers provided by Salesforce.

**Action Class**

You can use `ApexPages.Action` to create an action method that you can use in a Visualforce custom controller or controller extension.
Namespace
ApexPages

Usage
For example, you could create a `saveOver` method on a controller extension that performs a custom save.

Instantiation
The following code snippet illustrates how to instantiate a new `ApexPages.Action` object that uses the save action:

```java
ApexPages.Action saveAction = new ApexPages.Action('{!save}');
```

IN THIS SECTION:
- **Action Constructors**
- **Action Methods**

Action Constructors
The following are constructors for `Action`.

IN THIS SECTION:
- **Action(action)**
  - Creates a new instance of the `ApexPages.Action` class using the specified action.

**Action(action)**
Creates a new instance of the `ApexPages.Action` class using the specified action.

Signature
```
public Action(String action)
```

Parameters
- `action`
  - Type: String
  - The action.

Action Methods
The following are methods for `Action`. All are instance methods.
IN THIS SECTION:

getExpression()
Returns the expression that is evaluated when the action is invoked.

invoke()
Invokes the action.

getExpression()

Returns the expression that is evaluated when the action is invoked.

Signature

public String getExpression()

Return Value

Type: String

invoke()

Invokes the action.

Signature

public System.PageReference invoke()

Return Value

Type: System.PageReference

Component Class

Represents a dynamic Visualforce component in Apex.

Namespace

ApexPages

Dynamic Component Properties

The following are properties for Component.

IN THIS SECTION:

childComponents
Returns a reference to the child components for the component.
expressions
Sets the content of an attribute using the expression language notation. The notation for this is
expressions.name_of_attribute.

facets
Sets the content of a facet to a dynamic component. The notation is facet.name_of_facet.

childComponents
Returns a reference to the child components for the component.

Signature
public List <ApexPages.Component> childComponents {get; set;}

Property Value
Type: List<ApexPages.Component>

Example
Component.Apex.PageBlockSection pageBlkSection = new
Component.Apex.PageBlockSection(title='dummy header');
pageBlk.childComponents.add(pageBlkSection);

expressions
Sets the content of an attribute using the expression language notation. The notation for this is expressions.name_of_attribute.

Signature
public String expressions {get; set;}

Property Value
Type: String

Example
Component.Apex.InputField inpFld = new
Component.Apex.InputField();
inpField.expressions.value = '{!Account.Name}';
inpField.expressions.id = '{!$User.FirstName}';

facets
Sets the content of a facet to a dynamic component. The notation is facet.name_of_facet.
**Signature**

```csharp
public String facets {get; set;}
```

**Property Value**

Type: `String`

**Usage**

Note: This property is only accessible by components that support facets.

**Example**

```csharp
Component.Apex.DataTable myDT = new Component.Apex.DataTable();
myDT.facets.footer = footer;
```

**IdeaStandardController Class**

IdeaStandardController objects offer Ideas-specific functionality in addition to what is provided by the StandardController.

**Namespace**

ApexPages

**Usage**

A method in the IdeaStandardController object is called by and operated on a particular instance of an IdeaStandardController.

Note: The IdeaStandardSetController and IdeaStandardController classes are currently available through a limited release program. For information on enabling these classes for your organization, contact your Salesforce representative.

In addition to the methods listed in this class, the IdeaStandardController class inherits all the methods associated with the StandardController class.

**Instantiation**

An IdeaStandardController object cannot be instantiated. An instance can be obtained through a constructor of a custom extension controller when using the standard ideas controller.
Example

The following example shows how an IdeaStandardController object can be used in the constructor for a custom list controller. This example provides the framework for manipulating the comment list data before displaying it on a Visualforce page.

```java
public class MyIdeaExtension {

    private final ApexPages.IdeaStandardController ideaController;

    public MyIdeaExtension(ApexPages.IdeaStandardController controller) {
        ideaController = (ApexPages.IdeaStandardController)controller;
    }

    public List<IdeaComment> getModifiedComments() {
        IdeaComment[] comments = ideaController.getCommentList();
        // modify comments here
        return comments;
    }
}
```

The following Visualforce markup shows how the IdeaStandardController example shown above can be used in a page. This page must be named `detailPage` for this example to work.

---

Note: For the Visualforce page to display the idea and its comments, in the following example you need to specify the ID of a specific idea (for example, `/apex/detailPage?id=<ideaID>`) whose comments you want to view.

```xml
<!-- page named detailPage -->
<apex:page standardController="Idea" extensions="MyIdeaExtension">
    <apex:pageBlock title="Idea Section">
        <ideas:detailOutputLink page="detailPage" ideaId="{{idea.id}}">{{idea.title}}</ideas:detailOutputLink>
        <br/><br/>
        <apex:outputText >{{idea.body}}</apex:outputText>
    </apex:pageBlock>
    <apex:pageBlock title="Comments Section">
        <apex:dataList var="a" value="{{modifiedComments}}" id="list">
            {!a.commentBody}
        </apex:dataList>
        <ideas:detailOutputLink page="detailPage" ideaId="{{idea.id}}" pageOffset="-1">Prev</ideas:detailOutputLink>
        <ideas:detailOutputLink page="detailPage" ideaId="{{idea.id}}" pageOffset="1">Next</ideas:detailOutputLink>
    </apex:pageBlock>
</apex:page>
```

SEE ALSO:

- StandardController Class
- IdeaStandardController Methods

The following are instance methods for IdeaStandardController.
IN THIS SECTION:

getCommentList()

Returns the list of read-only comments from the current page.

**getCommentList()**

Returns the list of read-only comments from the current page.

**Signature**

```java
public IdeaComment[] getCommentList()
```

**Return Value**

Type: `IdeaComment[]`

This method returns the following comment properties:

- `id`
- `commentBody`
- `createdDate`
- `createdBy.Id`
- `createdBy.communityNickname`

**IdeaStandardSetController Class**

**IdeaStandardSetController** objects offer ideas-specific functionality in addition to what is provided by the **StandardSetController**.

**Namespace**

`ApexPages`

**Usage**

**Note:** The **IdeaStandardSetController** and **IdeaStandardController** classes are currently available through a limited release program. For information on enabling these classes for your organization, contact your Salesforce representative.

In addition to the method listed above, the **IdeaStandardSetController** class inherits the methods associated with the **StandardSetController**.

**Note:** The methods inherited from the **StandardSetController** cannot be used to affect the list of ideas returned by the **getIdeaList** method.

**Instantiation**

An **IdeaStandardSetController** object cannot be instantiated. An instance can be obtained through a constructor of a custom extension controller when using the standard list controller for ideas.
Example: Displaying a Profile Page

The following example shows how an IdeaStandardSetController object can be used in the constructor for a custom list controller:

```java
public class MyIdeaProfileExtension {
    private final ApexPages.IdeaStandardSetController ideaSetController;

    public MyIdeaProfileExtension(ApexPages.IdeaStandardSetController controller) {
        ideaSetController = (ApexPages.IdeaStandardSetController)controller;
    }

    public List<Idea> getModifiedIdeas() {
        Idea[] ideas = ideaSetController.getIdeaList();
        // modify ideas here
        return ideas;
    }
}
```

The following Visualforce markup shows how the IdeaStandardSetController example shown above and the `<ideas:profileListOutputLink>` component can display a profile page that lists the recent replies, submitted ideas, and votes associated with a user. Because this example does not identify a specific user ID, the page automatically shows the profile page for the current logged in user. This page must be named `profilePage` in order for this example to work:

```visualforce
<!-- page named profilePage -->
<apex:page standardController="Idea" extensions="MyIdeaProfileExtension"
    recordSetVar="ideaSetVar">
    <apex:pageBlock>
        <ideas:profileListOutputLink sort="recentReplies" page="profilePage">Recent Replies</ideas:profileListOutputLink>
        |<ideas:profileListOutputLink sort="ideas" page="profilePage">Ideas Submitted</ideas:profileListOutputLink>
        |<ideas:profileListOutputLink sort="votes" page="profilePage">Ideas Voted</ideas:profileListOutputLink>
    </apex:pageBlock>

    <apex:pageBlock>
        <apex:dataList value="{!modifiedIdeas}" var="ideadata">
            <ideas:detailoutputlink ideaId="{!ideadata.id}" page="viewPage">{!ideadata.title}</ideas:detailoutputlink>
        </apex:dataList>
    </apex:pageBlock>
</apex:page>
```

In the previous example, the `<ideas:detailoutputlink>` component links to the following Visualforce markup that displays the detail page for a specific idea. This page must be named `viewPage` in order for this example to work:

```visualforce
<!-- page named viewPage -->
<apex:page standardController="Idea">
    <apex:pageBlock title="Idea Section">
        <ideas:detailOutputLink page="viewPage" ideaId="{!idea.id}">{!idea.title}</ideas:detailOutputLink>
    </apex:pageBlock>
</apex:page>
```
Example: Displaying a List of Top, Recent, and Most Popular Ideas and Comments

The following example shows how an IdeaStandardSetController object can be used in the constructor for a custom list controller:

```java
public class MyIdeaListExtension {
    private final ApexPages.IdeaStandardSetController ideaSetController;

    public MyIdeaListExtension (ApexPages.IdeaStandardSetController controller) {
        ideaSetController = (ApexPages.IdeaStandardSetController)controller;
    }

    public List<Idea> getModifiedIdeas() {
        Idea[] ideas = ideaSetController.getIdeaList();
        // modify ideas here
        return ideas;
    }
}
```

The following Visualforce markup shows how the IdeaStandardSetController example shown above can be used with the `<ideas:listOutputLink>` component to display a list of recent, top, and most popular ideas and comments. This page must be named `listPage` in order for this example to work:

```xml
<apex:page standardController="Idea" extensions="MyIdeaListExtension"
     recordSetVar="ideaSetVar">
    <apex:pageBlock >
        <ideas:listOutputLink sort="recent" page="listPage">Recent Ideas</ideas:listOutputLink>
        |<ideas:listOutputLink sort="top" page="listPage">Top Ideas</ideas:listOutputLink>
        |<ideas:listOutputLink sort="popular" page="listPage">Popular Ideas</ideas:listOutputLink>
        |<ideas:listOutputLink sort="comments" page="listPage">Recent Comments</ideas:listOutputLink>
    </apex:pageBlock>
</apex:page>
```
In the previous example, the `<ideas:detailOutputLink>` component links to the following Visualforce markup that displays the detail page for a specific idea. This page must be named `viewPage`.

```
<!-- page named viewPage -->
<apex:page standardController="Idea">
    <apex:pageBlock title="Idea Section">
        <ideas:detailOutputLink page="viewPage" ideaId="{!idea.id}">{!idea.title}
        <br/>
        <apex:outputText>{!idea.body}</apex:outputText>
    </apex:pageBlock>
</apex:page>
```

SEE ALSO:
- StandardSetController Class

IdeaStandardSetController Methods

The following are instance methods for `IdeaStandardSetController`.

IN THIS SECTION:
- `getIdeaList()`

Returns the list of read-only ideas in the current page set.

**getIdeaList()**

Returns the list of read-only ideas in the current page set.

**Signature**

```
public Idea[] getIdeaList()
```

**Return Value**

Type: Idea[]

**Usage**

You can use the `<ideas:listOutputLink>`, `<ideas:profileListOutputLink>`, and `<ideas:detailOutputLink>` components to display profile pages as well as idea list and detail pages (see the examples below). The following is a list of properties returned by this method:

- Body
- Categories
- Category
- CreatedBy.CommunityNickname
- CreatedBy.Id
- CreatedDate
KnowledgeArticleVersionStandardController Class

KnowledgeArticleVersionStandardController objects offer article-specific functionality in addition to what is provided by the StandardController.

Namespace

ApexPages

Usage

In addition to the method listed above, the KnowledgeArticleVersionStandardController class inherits all the methods associated with StandardController.

Note: Though inherited, the edit, delete, and save methods don’t serve a function when used with the KnowledgeArticleVersionStandardController class.

Example

The following example shows how a KnowledgeArticleVersionStandardController object can be used to create a custom extension controller. In this example, you create a class named AgentContributionArticleController that allows customer-support agents to see pre-populated fields on the draft articles they create while closing cases.

Prerequisites:

1. Create an article type called FAQ. For instructions, see “Create Article Types” in the Salesforce online help.
2. Create a text custom field called Details. For instructions, see “Add Custom Fields to Article Types” in the Salesforce online help.
3. Create a category group called Geography and assign it to a category called USA. For instructions, see “Create and Modify Category Groups” and “Add Data Categories to Category Groups” in the Salesforce online help.
4. Create a category group called Topics and assign it a category called Maintenance.

```java
/** Custom extension controller for the simplified article edit page that appears when an article is created on the close-case page. */
public class AgentContributionArticleController {
    // The constructor must take a ApexPages.KnowledgeArticleVersionStandardController as an argument
```
public AgentContributionArticleController(
    ApexPages.KnowledgeArticleVersionStandardController ctl) {
    // This is the SObject for the new article.
    // It can optionally be cast to the proper article type.
    // For example, FAQ__kav article = (FAQ__kav) ctl.getRecord();
    SObject article = ctl.getRecord();
    // This returns the ID of the case that was closed.
    String sourceId = ctl.getSourceId();
    Case c = [SELECT Subject, Description FROM Case WHERE Id=:sourceId];

    // This overrides the default behavior of pre-filling the
    // title of the article with the subject of the closed case.
    article.put('title', 'From Case: ' + c.subject);
    article.put('details__c', c.description);

    // Only one category per category group can be specified.
    ctl.selectDataCategory('Geography','USA');
    ctl.selectDataCategory('Topics','Maintenance');
}

/** Test class for the custom extension controller. */
@isTest
private class AgentContributionArticleControllerTest {
    static testMethod void testAgentContributionArticleController() {
        String caseSubject = 'my test';
        String caseDesc = 'my test description';

        Case c = new Case();
        c.subject= caseSubject;
        c.description = caseDesc;
        insert c;
        String caseId = c.id;
        System.debug('Created Case: ' + caseId);

        ApexPages.currentPage().getParameters().put('sourceId', caseId);
        ApexPages.currentPage().getParameters().put('sfdc.override', '1');

        ApexPages.KnowledgeArticleVersionStandardController ctl =
            new ApexPages.KnowledgeArticleVersionStandardController(new FAQ__kav());

        new AgentContributionArticleController(ctl);

        System.assertEquals(caseId, ctl.getSourceId());
        System.assertEquals('From Case: ' + caseSubject, ctl.getRecord().get('title'));
        System.assertEquals(caseDesc, ctl.getRecord().get('details__c'));
    }
}

If you created the custom extension controller for the purpose described in the previous example (that is, to modify submitted-via-case articles), complete the following steps after creating the class:

1. Log into your Salesforce organization and from Setup, enter Knowledge Settings in the Quick Find box, then select Knowledge Settings.
2. Click **Edit**.

3. Assign the class to the Use Apex customization field. This associates the article type specified in the new class with the article type assigned to closed cases.

4. Click **Save**.

IN THIS SECTION:

KnowledgeArticleVersionStandardController Constructors

KnowledgeArticleVersionStandardController Methods

SEE ALSO:

StandardController Class

**KnowledgeArticleVersionStandardController Constructors**

The following are constructors for **KnowledgeArticleVersionStandardController**.

IN THIS SECTION:

**KnowledgeArticleVersionStandardController(article)**

Creates a new instance of the **ApexPages.KnowledgeArticleVersionStandardController** class using the specified knowledge article.

**KnowledgeArticleVersionStandardController (article)**

Creates a new instance of the **ApexPages.KnowledgeArticleVersionStandardController** class using the specified knowledge article.

**Signature**

```
public KnowledgeArticleVersionStandardController(SObject article)
```

**Parameters**

- **article**
  - Type: SObject
  - The knowledge article, such as FAQ_kav.

**KnowledgeArticleVersionStandardController Methods**

The following are instance methods for **KnowledgeArticleVersionStandardController**.

IN THIS SECTION:

**getSourceId()**

Returns the ID for the source object record when creating a new article from another object.
**setDataCategory(categoryGroup, category)**

Specifies a default data category for the specified data category group when creating a new article.

**getSourceId()**

Returns the ID for the source object record when creating a new article from another object.

**Signature**

```
public String getSourceId()
```

**Return Value**

Type: String

**setDataCategory(categoryGroup, category)**

Specifies a default data category for the specified data category group when creating a new article.

**Signature**

```
public Void setDataCategory(String categoryGroup, String category)
```

**Parameters**

- **categoryGroup**
  
  Type: String

- **category**
  
  Type: String

**Return Value**

Type: Void

**Message Class**

Contains validation errors that occur when the user saves the page that uses a standard controller.

**Namespace**

ApexPages

**Usage**

When using a standard controller, all validation errors, both custom and standard, that occur when the user saves the page are automatically added to the page error collections. If an inputField component is bound to the field with an error, the message is added to the component’s error collection. All messages are added to the page’s error collection. For more information, see Validation Rules and Standard Controllers in the Visualforce Developer’s Guide.

If your application uses a custom controller or extension, you must use the message class for collecting errors.
Instantiation

In a custom controller or controller extension, you can instantiate a Message in one of these ways:

- `ApexPages.Message myMsg = new ApexPages.Message(ApexPages.severity, summary);`
  
  where `ApexPages.severity` is the enum that determines how severe a message is, and `summary` is the String used to summarize the message. For example:
  
  ```java
  ```

- `ApexPages.Message myMsg = new ApexPages.Message(ApexPages.severity, summary, detail);`
  
  where `ApexPages.severity` is the enum that determines how severe a message is, `summary` is the String used to summarize the message, and `detail` is the String used to provide more detailed information about the error.

ApexPages.Severity Enum

To specify the severity of the message, use the `ApexPages.Severity` enum values. The following are the valid values:

- CONFIRM
- ERROR
- FATAL
- INFO
- WARNING

All enums have access to standard methods, such as `name` and `value`.

IN THIS SECTION:
- Message Constructors
- Message Methods

Message Constructors

The following are constructors for `Message`.

IN THIS SECTION:
- `Message(severity, summary)`
  Creates a new instance of the `ApexPages.Message` class using the specified message severity and summary.
- `Message(severity, summary, detail)`
  Creates a new instance of the `ApexPages.Message` class using the specified message severity, summary, and message detail.
- `Message(severity, summary, detail, id)`
  Creates a new instance of the `ApexPages.Message` class using the specified severity, summary, detail, and component ID.

**Message(severity, summary)**

Creates a new instance of the `ApexPages.Message` class using the specified message severity and summary.
Signature

```java
public Message(ApexPages.Severity severity, String summary)
```

Parameters

- **severity**
  - Type: `ApexPages.Severity`
  - The severity of a Visualforce message.

- **summary**
  - Type: `String`
  - The summary Visualforce message.

### Message(severity, summary, detail)

Creates a new instance of the `ApexPages.Message` class using the specified message severity, summary, and message detail.

Signature

```java
public Message(ApexPages.Severity severity, String summary, String detail)
```

Parameters

- **severity**
  - Type: `ApexPages.Severity`
  - The severity of a Visualforce message.

- **summary**
  - Type: `String`
  - The summary Visualforce message.

- **detail**
  - Type: `String`
  - The detailed Visualforce message.

### Message(severity, summary, detail, id)

Creates a new instance of the `ApexPages.Message` class using the specified severity, summary, detail, and component ID.

Signature

```java
public Message(ApexPages.Severity severity, String summary, String detail, String id)
```

Parameters

- **severity**
  - Type: `ApexPages.Severity`
  - The severity of a Visualforce message.
summary
Type: String
The summary Visualforce message.

detail
Type: String
The detailed Visualforce message.

id
Type: String
The ID of the Visualforce component to associate with the message, for example, a form field with an error.

Message Methods
The following are methods for Message. All are instance methods.

IN THIS SECTION:

getComponentLabel()
Returns the label of the associated inputField component. If no label is defined, this method returns null.

getDetail()
Returns the value of the detail parameter used to create the message. If no detail String was specified, this method returns null.

getSeverity()
Returns the severity enum used to create the message.

getSummary()
Returns the summary String used to create the message.

getComponentLabel()
Returns the label of the associated inputField component. If no label is defined, this method returns null.

Signature
public String getComponentLabel()

Return Value
Type: String

getDetail()
Returns the value of the detail parameter used to create the message. If no detail String was specified, this method returns null.

Signature
public String getDetail()
**Return Value**
Type: String

**getSeverity()**
Returns the severity enum used to create the message.

**Signature**
```java
public ApexPages.Severity getSeverity()
```

**Return Value**
Type: ApexPages.Severity

**getSummary()**
Returns the summary String used to create the message.

**Signature**
```java
public String getSummary()
```

**Return Value**
Type: String

---

**StandardController Class**

Use a StandardController when defining an extension for a standard controller.

**Namespace**
**ApexPages**

**Usage**
StandardController objects reference the pre-built Visualforce controllers provided by Salesforce. The only time it is necessary to refer to a StandardController object is when defining an extension for a standard controller. StandardController is the data type of the single argument in the extension class constructor.

**Instantiation**
You can instantiate a StandardController in the following way:

```java
ApexPages.StandardController sc = new ApexPages.StandardController(sObject);
```
Example

The following example shows how a StandardController object can be used in the constructor for a standard controller extension:

```java
public class myControllerExtension {

    private final Account acct;

    // The extension constructor initializes the private member
    // variable acct by using the getRecord method from the standard
    // controller.
    public myControllerExtension(ApexPages.StandardController stdController) {
        this.acct = (Account)stdController.getRecord();
    }

    public String getGreeting() {
        return 'Hello ' + acct.name + ' (' + acct.id + ')';
    }
}
```

The following Visualforce markup shows how the controller extension from above can be used in a page:

```xml
<apex:page standardController="Account" extensions="myControllerExtension">
    {!greeting} <p/>
    <apex:form>
        <apex:inputField value="{!account.name}"/> <p/>
        <apex:commandButton value="Save" action="{!save}"/>
    </apex:form>
</apex:page>
```

IN THIS SECTION:
StandardController Constructors
StandardController Methods

StandardController Constructors

The following are constructors for StandardController.

IN THIS SECTION:
StandardController(controllerSObject)
    Creates a new instance of the ApexPages.StandardController class for the specified standard or custom object.

**StandardController** (controllerSObject)

Creates a new instance of the ApexPages.StandardController class for the specified standard or custom object.

**Signature**

```java
public StandardController(SObject controllerSObject)
```
Parameters

controllerSObject
Type: SObject
A standard or custom object.

StandardController Methods

The following are methods for StandardController. All are instance methods.

IN THIS SECTION:

addFields(fieldNames)
When a Visualforce page is loaded, the fields accessible to the page are based on the fields referenced in the Visualforce markup. This method adds a reference to each field specified in fieldNames so that the controller can explicitly access those fields as well.

cancel()
Returns the PageReference of the cancel page.

delete()
Deletes record and returns the PageReference of the delete page.

edit()
Returns the PageReference of the standard edit page.

getId()
Returns the ID of the record that is currently in context, based on the value of the id query string parameter in the Visualforce page URL.

getRecord()
Returns the record that is currently in context, based on the value of the id query string parameter in the Visualforce page URL.

reset()
Forces the controller to reacquire access to newly referenced fields. Any changes made to the record prior to this method call are discarded.

save()
Saves changes and returns the updated PageReference.

view()
Returns the PageReference object of the standard detail page.

addFields(fieldNames)
When a Visualforce page is loaded, the fields accessible to the page are based on the fields referenced in the Visualforce markup. This method adds a reference to each field specified in fieldNames so that the controller can explicitly access those fields as well.

Signature

public Void addFields(List<String> fieldNames)
Parameters
fieldNames
Type: List<String>

Return Value
Type: Void

Usage
This method should be called before a record has been loaded—typically, it's called by the controller's constructor. If this method is called outside of the constructor, you must use the reset() method before calling addFields().
The strings in fieldNames can either be the API name of a field, such as AccountId, or they can be explicit relationships to fields, such as something__r.myField__c.
This method is only for controllers used by dynamicVisualforce bindings.

cancel()
Returns the PageReference of the cancel page.

Signature
public System.PageReference cancel()

Return Value
Type: System.PageReference

delete()
Deletes record and returns the PageReference of the delete page.

Signature
public System.PageReference delete()

Return Value
Type: System.PageReference

edit()
Returns the PageReference of the standard edit page.

Signature
public System.PageReference edit()
Return Value
Type: System.PageReference

**getId()**
Returns the ID of the record that is currently in context, based on the value of the `id` query string parameter in the Visualforce page URL.

**Signature**
```
public String getId()
```

Return Value
Type: String

**getRecord()**
Returns the record that is currently in context, based on the value of the `id` query string parameter in the Visualforce page URL.

**Signature**
```
public SObject getRecord()
```

Return Value
Type: sObject

**Usage**
Note that only the fields that are referenced in the associated Visualforce markup are available for querying on this SObject. All other fields, including fields from any related objects, must be queried using a SOQL expression.

⚠️ **Tip:** You can work around this restriction by including a hidden component that references any additional fields that you want to query. Hide the component from display by setting the component’s `rendered` attribute to `false`.

**Example**
```
<apex:outputText
 value="{!account.billingcity}
 {!account.contacts}"
 rendered="false"/>
```

**reset()**
Forces the controller to reacquire access to newly referenced fields. Any changes made to the record prior to this method call are discarded.
Signature
public Void reset()

Return Value
Type: Void

Usage
This method is only used if addFields is called outside the constructor, and it must be called directly before addFields. This method is only for controllers used by dynamicVisualforce bindings.

save()
Saves changes and returns the updated PageReference.

Signature
public System.PageReference save()

Return Value
Type: System.PageReference

view()
Returns the PageReference object of the standard detail page.

Signature
public System.PageReference view()

Return Value
Type: System.PageReference

StandardSetController Class
StandardSetController objects allow you to create list controllers similar to, or as extensions of, the pre-built Visualforce list controllers provided by Salesforce.

Namespace
ApexPages

Usage
The StandardSetController class also contains a prototype object. This is a single sObject contained within the Visualforce StandardSetController class. If the prototype object’s fields are set, those values are used during the save action, meaning that the values
are applied to every record in the set controller’s collection. This is useful for writing pages that perform mass updates (applying identical changes to fields within a collection of objects).

Note: Fields that are required in other Salesforce objects will keep the same requiredness when used by the prototype object.

Instantiation

You can instantiate a StandardSetController in either of the following ways:

- From a list of sObjects:

```java
List<account> accountList = [SELECT Name FROM Account LIMIT 20];
ApexPages.StandardSetController ssc = new ApexPages.StandardSetController(accountList);
```

- From a query locator:

```java
ApexPages.StandardSetController ssc = new ApexPages.StandardSetController(Database.getQueryLocator([SELECT Name, CloseDate FROM Opportunity]));
```

Note: The maximum record limit for StandardSetController is 10,000 records. Instantiating StandardSetController using a query locator returning more than 10,000 records causes a LimitException to be thrown. However, instantiating StandardSetController with a list of more than 10,000 records doesn’t throw an exception, and instead truncates the records to the limit.

Example

The following example shows how a StandardSetController object can be used in the constructor for a custom list controller:

```java
public class opportunityList2Con {
    // ApexPages.StandardSetController must be instantiated
    // for standard list controllers
    public ApexPages.StandardSetController setCon {
        get {
            if(setCon == null) {
                setCon = new ApexPages.StandardSetController(Database.getQueryLocator([SELECT Name, CloseDate FROM Opportunity]));
            }
            return setCon;
        }
        set;
    }

    // Initialize setCon and return a list of records
    public List<Opportunity> getOpportunities() {
        return (List<Opportunity>) setCon.getRecords();
    }
}
```

The following Visualforce markup shows how the controller above can be used in a page:

```xml
<apex:page controller="opportunityList2Con">
    <apex:pageBlock>
        <apex:pageBlockTable value="{"opportunities}" var="o">
            <apex:column value="{!o.Name}="/>
        </apex:pageBlockTable>
    </apex:pageBlock>
</apex:page>
```
IN THIS SECTION:

StandardSetController Constructors
StandardSetController Methods

StandardSetController Constructors

The following are constructors for StandardSetController.

IN THIS SECTION:

StandardSetController(queryLocator)
Creates an instance of the ApexPages.StandardSetController class for the list of objects returned by the query locator.

Signature

public StandardSetController(Database.QueryLocator queryLocator)

Parameters

queryLocator
Type: Database.QueryLocator
A query locator representing a list of sObjects.

StandardSetController(controllerSObjects)
Creates an instance of the ApexPages.StandardSetController class for the specified list of standard or custom objects.

Signature

public StandardSetController(List<sObject> controllerSObjects)

Parameters

controllerSObjects
Type: List on page 3177<sObject on page 3372>
A List of standard or custom objects.
Example

```java
List<account> accountList = [SELECT Name FROM Account LIMIT 20];
ApexPages.StandardSetController ssc = new ApexPages.StandardSetController(accountList);
```

**StandardSetController Methods**

The following are methods for `StandardSetController`. All are instance methods.

**IN THIS SECTION:**

- `cancel()`: Returns the `PageReference` of the original page, if known, or the home page.
- `first()`: Returns the first page of records.
- `getCompleteResult()`: Indicates whether there are more records in the set than the maximum record limit. If this is false, there are more records than you can process using the list controller. The maximum record limit is 10,000 records.
- `getFilterId()`: Returns the ID of the filter that is currently in context.
- `getHasNext()`: Indicates whether there are more records after the current page set.
- `getHasPrevious()`: Indicates whether there are more records before the current page set.
- `getListViewOptions()`: Returns a list of the listviews available to the current user.
- `getPageSize()`: Returns the number of records included in each page set.
- `getRecord()`: Returns the `sObject` that represents the changes to the selected records. This retrieves the prototype object contained within the class, and is used for performing mass updates.
- `getRecords()`: Returns the list of `sObjects` in the current page set. This list is immutable, i.e. you can't call `clear()` on it.
- `getResultSize()`: Returns the number of records in the set.
- `getSelected()`: Returns the list of `sObjects` that have been selected.
- `last()`: Returns the last page of records.
- `next()`: Returns the next page of records.
previous()
Returns the previous page of records.

save()
Inserts new records or updates existing records that have been changed. After this operation is finished, it returns a PageReference
to the original page, if known, or the home page.

setFilterID(filterId)
Sets the filter ID of the controller.

setpageNumber(pageNumber)
Sets the page number.

setPageSize(pageSize)
Sets the number of records in each page set.

setSelected(selectedRecords)
Set the selected records.

cancel()
Returns the PageReference of the original page, if known, or the home page.

Signature
public System.PageReference cancel()

Return Value
Type: System.PageReference

first()
Returns the first page of records.

Signature
public Void first()

Return Value
Type: Void

getCompleteResult()
Indicates whether there are more records in the set than the maximum record limit. If this is false, there are more records than you can
process using the list controller. The maximum record limit is 10,000 records.

Signature
public Boolean getCompleteResult()
Return Value
Type: Boolean

`getFilterId()`
Returns the ID of the filter that is currently in context.

Signature
`public String getFilterId()`

Return Value
Type: String

`getHasNext()`
Indicates whether there are more records after the current page set.

Signature
`public Boolean getHasNext()`

Return Value
Type: Boolean

`getHasPrevious()`
Indicates whether there are more records before the current page set.

Signature
`public Boolean getHasPrevious()`

Return Value
Type: Boolean

`getListViewOptions()`
Returns a list of the listviews available to the current user.

Signature
`public System.SelectOption[] getListViewOptions()`

Return Value
Type: `System.SelectOption[]`
**getPageNumber()**
Returns the page number of the current page set. Note that the first page returns 1.

**Signature**
`public Integer getPageNumber()`

**Return Value**
Type: Integer

**getPageSize()**
Returns the number of records included in each page set.

**Signature**
`public Integer getPageSize()`

**Return Value**
Type: Integer

**getRecord()**
Returns the sObject that represents the changes to the selected records. This retrieves the prototype object contained within the class, and is used for performing mass updates.

**Signature**
`public sObject getRecord()`

**Return Value**
Type: sObject

**getRecords()**
Returns the list of sObjects in the current page set. This list is immutable, i.e. you can't call `clear()` on it.

**Signature**
`public sObject[] getRecords()`

**Return Value**
Type: sObject[]
**getResultSize()**

Returns the number of records in the set.

**Signature**

```java
public Integer getResultSize()
```

**Return Value**

Type: `Integer`

**getSelected()**

Returns the list of sObjects that have been selected.

**Signature**

```java
public sObject[] getSelected()
```

**Return Value**

Type: `sObject[]`

**last()**

Returns the last page of records.

**Signature**

```java
public Void last()
```

**Return Value**

Type: `Void`

**next()**

Returns the next page of records.

**Signature**

```java
public Void next()
```

**Return Value**

Type: `Void`

**previous()**

Returns the previous page of records.
Signature

public Void previous()

Return Value
Type: Void

save()
Inserts new records or updates existing records that have been changed. After this operation is finished, it returns a PageReference to the original page, if known, or the home page.

Signature

public System.PageReference save()

Return Value
Type: System.PageReference

setFilterID(filterId)
Sets the filter ID of the controller.

Signature

public Void setFilterID(String filterId)

Parameters
filterId
Type: String

Return Value
Type: Void

setpageNumber(pageNumber)
Sets the page number.

Signature

public Void setPageNumber(Integer pageNumber)

Parameters
pageNumber
Type: Integer
Return Value
Type: Void

**setPageSize(pageSize)**
Sets the number of records in each page set.

**Signature**
```
public Void setPageSize(Integer pageSize)
```

**Parameters**
- **pageSize**
  Type: Integer

Return Value
Type: Void

**setSelected(selectedRecords)**
Set the selected records.

**Signature**
```
public Void setSelected(sObject[] selectedRecords)
```

**Parameters**
- **selectedRecords**
  Type: sObject[]

Return Value
Type: Void

---

**AppLauncher Namespace**

The AppLauncher namespace provides methods for managing the appearance of apps in the App Launcher, including their visibility and sort order.

The following class is in the AppLauncher namespace.

**IN THIS SECTION:**
- **AppMenu Class**
  Contains methods to set the appearance of apps in the App Launcher.
AppMenu Class

Contains methods to set the appearance of apps in the App Launcher.

Namespace

AppLauncher

IN THIS SECTION:

AppMenu Methods

AppMenu Methods

The following are methods for AppMenu.

IN THIS SECTION:

setAppVisibility(appMenuItemId, isVisible)
Shows or hides specific apps in the App Launcher.

setOrgSortOrder(appIds)
Sets the organization-wide default sort order for the App Launcher based on a List of app menu item IDs in the desired order.

setUserSortOrder(appIds)
Sets an individual user's default sort order for the App Launcher based on a List of app menu item IDs in the desired order.

**setAppVisibility(appMenuItemId, isVisible)**

Shows or hides specific apps in the App Launcher.
Signature

```java
public static void setAppVisibility(Id appMenuItemId, Boolean isVisible)
```

Parameters

- **appMenuItemId**
  - Type: Id
  - The 15-character application ID value for an app. For more information, see the ApplicationId field for `AppMenuItem` or the AppMenuItemId field for `UserAppMenuItem` in the Salesforce Object Reference.

- **isVisible**
  - Type: Boolean
  - If `true`, the app is visible.

Return Value

Type: void

**setOrgSortOrder(appIds)**

Sets the organization-wide default sort order for the App Launcher based on a List of app menu item IDs in the desired order.

Signature

```java
public static void setOrgSortOrder(List<Id> appIds)
```

Parameters

- **appIds**
  - Type: List<Id>
  - A list of application ID values. For more information, see the ApplicationId field for `AppMenuItem` in the Salesforce Object Reference.

Return Value

Type: void

**setUserSortOrder(appIds)**

Sets an individual user’s default sort order for the App Launcher based on a List of app menu item IDs in the desired order.

Signature

```java
public static void setUserSortOrder(List<Id> appIds)
```

Parameters

- **appIds**
  - Type: List<Id>
A list of application ID values. For more information, see the `AppMenuItemId` field for `UserAppMenuItem` in the `Salesforce Object Reference`.

Return Value
Type: `void`

**ChangePasswordController Class**
This class and its methods are for internal use only.

**Namespace**
`AppLauncher`

**CommunityLogoController Class**
This class and its methods are for internal use only.

**Namespace**
`AppLauncher`

**EmployeeLoginLinkController Class**
This class and its methods are for internal use only.

**Namespace**
`AppLauncher`

**ForgotPasswordController Class**
This class and its methods are for internal use only.

**Namespace**
`AppLauncher`

**IdentityHeaderController Class**
This class and its methods are for internal use only.

**Namespace**
`AppLauncher`
LoginFormController Class

This class and its methods are for internal use only.

Namespace

AppLauncher

SelfRegisterController Class

This class and its methods are for internal use only.

Namespace

AppLauncher

SocialLoginController Class

This class and its methods are for internal use only.

Namespace

AppLauncher

Approval Namespace

The Approval namespace provides classes and methods for approval processes.

The following are the classes in the Approval namespace.

IN THIS SECTION:

LockResult Class
The result of a record lock returned by a System.Approval.lock() method.

ProcessRequest Class
The ProcessRequest class is the parent class for the ProcessSubmitRequest and ProcessWorkitemRequest classes. Use the ProcessRequest class to write generic Apex that can process objects from either class.

ProcessResult Class
After you submit a record for approval, use the ProcessResult class to process the results of an approval process.

ProcessSubmitRequest Class
Use the ProcessSubmitRequest class to submit a record for approval.

ProcessWorkitemRequest Class
Use the ProcessWorkitemRequest class for processing an approval request after it is submitted.

UnlockResult Class
The result of a record unlock, returned by a System.Approval.unlock() method.
LockResult Class

The result of a record lock returned by a System.Approval.lock() method.

Namespace

Approval

Usage

The System.Approval.lock() methods return Approval.LockResult objects. Each element in a LockResult array corresponds to an element in the ID or sObject array passed as a parameter to a lock method. The first element in the LockResult array corresponds to the first element in the ID or sObject array, the second element corresponds to the second element, and so on. If only one ID or sObject is passed in, the LockResult array contains a single element.

Example

The following example obtains and iterates through the returned Approval.LockResult objects. It locks some queried accounts using Approval.lock with a false second parameter to allow partial processing of records on failure. Next, it iterates through the results to determine whether the operation was successful for each record. It writes the ID of every record that was processed successfully to the debug log, or writes error messages and failed fields of the failed records.

```
// Query the accounts to lock
Account[] accts = [SELECT Id from Account WHERE Name LIKE 'Acme%'];
// Lock the accounts
Approval.LockResult[] lrList = Approval.lock(accts, false);

// Iterate through each returned result
for(Approval.LockResult lr : lrList) {
    if (lr.isSuccess()) {
        // Operation was successful, so get the ID of the record that was processed
        System.debug('Successfully locked account with ID: ' + lr.getId());
    } else {
        // Operation failed, so get all errors
        for(Database.Error err : lr.getErrors()) {
            System.debug('The following error has occurred: ');
            System.debug(err.getStatusCode() + ': ' + err.getMessage());
            System.debug('Account fields that affected this error: ' + err.getFields());
        }
    }
}
```

IN THIS SECTION:

LockResult Methods

SEE ALSO:

Approval Class
LockResult Methods

The following are methods for LockResult.

IN THIS SECTION:
  * getErrors()
    * If an error occurred, returns an array of one or more database error objects, providing the error code and description.
  * getId()
    * Returns the ID of the sObject you are trying to lock.
  * isSuccess()
    * A Boolean value that is set to true if the lock operation is successful for this object, or false otherwise.

getErrors()

If an error occurred, returns an array of one or more database error objects, providing the error code and description.

Signature

public List<Database.Error> getErrors()

Return Value

Type: List<Database.Error>

getId()

Returns the ID of the sObject you are trying to lock.

Signature

public Id getId()

Return Value

Type: Id

Usage

If the field contains a value, the object was locked. If the field is empty, the operation was not successful.

isSuccess()

A Boolean value that is set to true if the lock operation is successful for this object, or false otherwise.

Signature

public Boolean isSuccess()
Return Value
Type: Boolean

ProcessRequest Class

The ProcessRequest class is the parent class for the ProcessSubmitRequest and ProcessWorkItemRequest classes. Use the ProcessRequest class to write generic Apex that can process objects from either class.

Namespace
Approval

Usage
The request must be instantiated via the child classes, ProcessSubmitRequest and ProcessWorkItemRequest.

ProcessRequest Methods
The following are methods for ProcessRequest. All are instance methods.

IN THIS SECTION:
getComments()  
Returns the comments that have been added previously to the approval request.
getNextApproverIds()  
Returns the list of user IDs of user specified as approvers.
setComments(comments)  
Sets the comments to be added to the approval request.
setNextApproverIds(nextApproverIds)  
If the next step in your approval process is another Apex approval process, you specify exactly one user ID as the next approver. If not, you cannot specify a user ID and this method must be null.

getComments()

Returns the comments that have been added previously to the approval request.

Signature
public String getComments()

Return Value
Type: String

g getNextApproverIds()

Returns the list of user IDs of user specified as approvers.
Signature

public ID[] getNextApproverIds()

Return Value

Type: ID[]

setComments(comments)

Sets the comments to be added to the approval request.

Signature

public Void setComments(String comments)

Parameters

comments

Type: String

Return Value

Type: Void

setNextApproverIds(nextApproverIds)

If the next step in your approval process is another Apex approval process, you specify exactly one user ID as the next approver. If not, you cannot specify a user ID and this method must be null.

Signature

public Void setNextApproverIds(ID[] nextApproverIds)

Parameters

nextApproverIds

Type: ID[]

Must be a single-entry list.

Return Value

Type: Void

ProcessResult Class

After you submit a record for approval, use the ProcessResult class to process the results of an approval process.
Namespace

Approval

Usage

A ProcessResult object is returned by the process method. You must specify the Approval namespace when creating an instance of this class. For example:

```java
Approval.ProcessResult result = Approval.process(req1);
```

ProcessResult Methods

The following are methods for ProcessResult. All are instance methods.

IN THIS SECTION:

- `getEntityId()`
  The ID of the record being processed.

- `getErrors()`
  If an error occurred, returns an array of one or more database error objects including the error code and description.

- `getInstanceId()`
  The ID of the approval process that has been submitted for approval.

- `getInstanceStatus()`
  The status of the current approval process. Valid values are: Approved, Rejected, Removed or Pending.

- `getNewWorkitemIds()`
  The IDs of the new items submitted to the approval process. There can be 0 or 1 approval processes.

- `isSuccess()`
  A Boolean value that is set to `true` if the approval process completed successfully; otherwise, it is set to `false`.

**getEntityId()**

The ID of the record being processed.

**Signature**

```java
public String getEntityId()
```

**Return Value**

Type: `String`

**getErrors()**

If an error occurred, returns an array of one or more database error objects including the error code and description.
Signature
public Database.Error[] getErrors()

Return Value
Type: Database.Error[]

getInstanceId()
The ID of the approval process that has been submitted for approval.

Signature
public String getInstanceId()

Return Value
Type: String

getInstanceStatus()
The status of the current approval process. Valid values are: Approved, Rejected, Removed or Pending.

Signature
public String getInstanceStatus()

Return Value
Type: String

getNewWorkitemIds()
The IDs of the new items submitted to the approval process. There can be 0 or 1 approval processes.

Signature
public ID[] getNewWorkitemIds()

Return Value
Type: ID[]

isSuccess()
A Boolean value that is set to true if the approval process completed successfully; otherwise, it is set to false.

Signature
public Boolean isSuccess()
Return Value
Type: Boolean

ProcessSubmitRequest Class
Use the ProcessSubmitRequest class to submit a record for approval.

Namespace
Approval

Usage
You must specify the Approval namespace when creating an instance of this class. The constructor for this class takes no arguments.
For example:

```
Approval.ProcessSubmitRequest psr = new Approval.ProcessSubmitRequest();
```

Inherited Methods
In addition to the methods listed, the ProcessSubmitRequest class has access to all the methods in its parent class, ProcessRequest Class.
- getComments()
- getNextApproverIds()
- setComments(comments)
- setNextApproverIds(nextApproverIds)

Example
To view sample code, refer to Approval Processing Example.

ProcessSubmitRequest Methods
The following are methods for ProcessSubmitRequest. All are instance methods.

IN THIS SECTION:
- getObjectId()
  Returns the ID of the record that has been submitted for approval. For example, it can return an account, contact, or custom object record.
- getProcessDefinitionNameOrId()
  Returns the developer name or ID of the process definition.
- getSkipEntryCriteria()
  If getProcessDefinitionNameOrId() returns a value other than null, getSkipEntryCriteria() determines whether to evaluate the entry criteria for the process (true) or not (false).
getSubmitterId()
Returns the user ID of the submitter requesting the approval record. The user must be one of the allowed submitters in the process definition setup.

setObjectId(recordId)
Sets the ID of the record to be submitted for approval. For example, it can specify an account, contact, or custom object record.

setProcessDefinitionNameOrId(nameOrId)
Sets the developer name or ID of the process definition to be evaluated.

setSkipEntryCriteria(skipEntryCriteria)
If the process definition name or ID is not null, setSkipEntryCriteria() determines whether to evaluate the entry criteria for the process (true) or not (false).

setSubmitterId(userID)
Sets the user ID of the submitter requesting the approval record. The user must be one of the allowed submitters in the process definition setup. If you don’t set a submitter ID, the process uses the current user as the submitter.

getObjectId()
Returns the ID of the record that has been submitted for approval. For example, it can return an account, contact, or custom object record.

Signature
class ProcessSubmitRequest
{
    public String getObjectId()
    {
    
    }
}

Return Value
Type: String

getProcessDefinitionNameOrId()
Returns the developer name or ID of the process definition.

Signature
class ProcessSubmitRequest
{
    public String getProcessDefinitionNameOrId()
    {
    
    }
}

Return Value
Type: String

Usage
The default is null. If the return value is null, when a user submits a record for approval Salesforce evaluates the entry criteria for all processes applicable to the user.

getSkipEntryCriteria()
If getProcessDefinitionNameOrId() returns a value other than null, getSkipEntryCriteria() determines whether to evaluate the entry criteria for the process (true) or not (false).
**getSkipEntryCriteria()**

Returns a boolean value indicating whether the entry criteria are being skipped.

**Signature**

```java
public Boolean getSkipEntryCriteria()
```

**Return Value**

Type: Boolean

**getSubmitterId()**

Returns the user ID of the submitter requesting the approval record. The user must be one of the allowed submitters in the process definition setup.

**Signature**

```java
public String getSubmitterId()
```

**Return Value**

Type: String

**setObjectId(recordId)**

Sets the ID of the record to be submitted for approval. For example, it can specify an account, contact, or custom object record.

**Signature**

```java
public Void setObjectId(String recordId)
```

**Parameters**

- `recordId`
  Type: String

**Return Value**

Type: Void

**setProcessDefinitionNameOrId(nameOrId)**

Sets the developer name or ID of the process definition to be evaluated.

**Signature**

```java
public Void setProcessDefinitionNameOrId(String nameOrId)
```

**Parameters**

- `nameOrId`
  Type: String
The process definition developer name or process definition ID. The record is submitted to this specific process. If set to `null`, submission of a record approval follows standard evaluation; that is, every entry criteria of the process definition in the process order is evaluated and the one that satisfies is picked and submitted.

**Return Value**
Type: Void

**Usage**
If the process definition name or ID is not set via this method, then by default it is null. If it is null, the submission of a record for approval evaluates entry criteria for all processes applicable to the submitter. The order of evaluation is based on the process order of the setup.

**setSkipEntryCriteria**
`skipEntryCriteria`
If the process definition name or ID is not null, `setSkipEntryCriteria()` determines whether to evaluate the entry criteria for the process (`true`) or not (`false`).

**Signature**
`public Void setSkipEntryCriteria(Boolean skipEntryCriteria)`

**Parameters**
`skipEntryCriteria`
Type: Boolean
- If set to `true`, request submission skips the evaluation of entry criteria for the process set in `setProcessDefinitionNameOrId(nameOrId)`.
- If the process definition name or ID is not specified, this parameter is ignored and standard evaluation is followed based on process order. If set to `false`, or if this method isn’t called, the entry criteria is not skipped.

**Return Value**
Type: Void

**setSubmitterId**
`userID`
Sets the user ID of the submitter requesting the approval record. The user must be one of the allowed submitters in the process definition setup. If you don’t set a submitter ID, the process uses the current user as the submitter.

**Signature**
`public Void setSubmitterId(String userID)`

**Parameters**
`userID`
Type: String
- The user ID on behalf of which the record is submitted. If set to `null`, the current user is the submitter. If the submitter is not set with this method, the default submitter is null (the current user).
Return Value
Type: Void

ProcessWorkitemRequest Class
Use the ProcessWorkitemRequest class for processing an approval request after it is submitted.

Namespace
Approval

Usage
You must specify the Approval namespace when creating an instance of this class. The constructor for this class takes no arguments. For example:

```
```

Inherited Methods
In addition to the methods listed, the ProcessWorkitemRequest class has access to all the methods in its parent class, ProcessRequest Class:

- getComments()
- getNextApproverIds()
- setComments(comments)
- setNextApproverIds(nextApproverIds)

ProcessWorkitemRequest Methods
The following are methods for ProcessWorkitemRequest. All are instance methods.

**getAction()**
Returns the type of action already associated with the approval request. Valid values are: Approve, Reject, or Removed.

**getWorkitemId()**
Returns the ID of the approval request that is in the process of being approved, rejected, or removed.

**setAction(actionType)**
Sets the type of action to take for processing an approval request.

**setWorkitemId(id)**
Sets the ID of the approval request that is being approved, rejected, or removed.
Signature
public String getAction()

Return Value
Type: String

getWorkitemId()
Returns the ID of the approval request that is in the process of being approved, rejected, or removed.

Signature
public String getWorkitemId()

Return Value
Type: String

setAction(actionType)
Sets the type of action to take for processing an approval request.

Signature
public Void setAction(String actionType)

Parameters
actionType
  Type: String
  Valid values are: Approve, Reject, or Removed. Only system administrators can specify Removed.

Return Value
Type: Void

setWorkitemId(id)
Sets the ID of the approval request that is being approved, rejected, or removed.

Signature
public Void setWorkitemId(String id)

Parameters
id
  Type: String
Return Value
Type: Void

UnlockResult Class
The result of a record unlock, returned by a System.Approval.unlock() method.

Namespace
Approval

Usage
The System.Approval.unlock() methods return Approval.UnlockResult objects. Each element in an UnlockResult array corresponds to an element in the ID or sObject array passed as a parameter to an unlock method. The first element in the UnlockResult array corresponds to the first element in the ID or sObject array, the second element corresponds to the second element, and so on. If only one ID or sObject is passed in, the UnlockResult array contains a single element.

Example
The following example shows how to obtain and iterate through the returned Approval.UnlockResult objects. It locks some queried accounts using Approval.unlock with a false second parameter to allow partial processing of records on failure. Next, it iterates through the results to determine whether the operation was successful for each record. It writes the ID of every record that was processed successfully to the debug log, or writes error messages and failed fields of the failed records.

```java
// Query the accounts to unlock
Account[] accts = [SELECT Id from Account WHERE Name LIKE 'Acme%'];

for(Account acct: accts) {
    // Create an approval request for the account
    Approval.ProcessSubmitRequest req1 = new Approval.ProcessSubmitRequest();
    req1.setComments('Submitting request for approval.');
    req1.setObjectId(acct.id);
    // Submit the record to specific process and skip the criteria evaluation
    req1.setProcessDefinitionNameOrId('PTO_Request_Process');
    req1.setSkipEntryCriteria(true);
    // Submit the approval request for the account
    Approval.ProcessResult result = Approval.process(req1);
    // Verify the result
    System.assert(result.isSuccess());
}
// Unlock the accounts
Approval.UnlockResult[] urList = Approval.unlock(accts, false);

// Iterate through each returned result
for(Approval.UnlockResult ur : urList) {
    // Process each UnlockResult object
}
```
if (ur.isSuccess()) {
    // Operation was successful, so get the ID of the record that was processed
    System.debug('Successfully unlocked account with ID: ' + ur.getId());
} else {
    // Operation failed, so get all errors
    for (Database.Error err : ur.getErrors()) {
        System.debug('The following error has occurred: ');
        System.debug(err.getStatusCode() + ': ' + err.getMessage());
        System.debug('Account fields that affected this error: ' + err.getFields());
    }
}

IN THIS SECTION:
UnlockResult Methods

SEE ALSO:
Approval Class

UnlockResult Methods
The following are methods for UnlockResult.

IN THIS SECTION:
getErrors()
If an error occurred, returns an array of one or more database error objects, providing the error code and description.

getId()
Returns the ID of the sObject you are trying to unlock.

isSuccess()
A Boolean value that is set to true if the unlock operation is successful for this object, or false otherwise.

getErrors()
If an error occurred, returns an array of one or more database error objects, providing the error code and description.

Signature
public List<Database.Error> getErrors()

Return Value
Type: List<Database.Error>

getId()
Returns the ID of the sObject you are trying to unlock.
Signature

```java
public Id getId()
```

Return Value

Type: Id

Usage

If the field contains a value, the object was unlocked. If the field is empty, the operation was not successful.

**isSuccess()**

A Boolean value that is set to `true` if the unlock operation is successful for this object, or `false` otherwise.

Signature

```java
public Boolean isSuccess()
```

Return Value

Type: Boolean

---

**Auth Namespace**

The `Auth` namespace provides an interface and classes for single sign-on into Salesforce and session security management. The following is the interface in the `Auth` namespace.

**IN THIS SECTION:**

- **AuthConfiguration Class**
  - Contains methods for configuring settings for users to log in to a Salesforce org using their authentication provider credentials instead of their Salesforce credentials. The authentication provider can be any authentication provider that supports the OpenID Connect protocol, such as Google, Facebook, or Twitter. Users log in to either an Experience Cloud site (https://MyDomainName.my.site.com) or your My Domain login URL (https://MyDomainName.my.salesforce.com).

- **AuthProviderCallbackState Class**
  - Provides request HTTP headers, body, and query parameters to the `AuthProviderPlugin.handleCallback` method for user authentication. This class allows you to group the information passed in rather than passing headers, body, and query parameters individually.

- **AuthProviderPlugin Interface**
  - This interface is deprecated. For new development, use the abstract class `Auth.AuthProviderPluginClass` to create a custom OAuth-based authentication provider plug-in for single sign-on in to Salesforce.

- **AuthProviderPluginClass**
  - Contains methods to create a custom OAuth-based authentication provider plug-in for single sign-on in to Salesforce. Use this class to create a custom authentication provider plug-in if you can’t use one of the authentication providers that Salesforce provides.
AuthProviderTokenResponse Class
Stores the response from the AuthProviderPlugin.handleCallback method.

AuthToken Class
Contains methods for providing the access token associated with an authentication provider for an authenticated user, except for the Janrain provider.

CommunitiesUtil Class
Contains methods for getting information about an Experience Cloud user.

ConfigurableSelfRegHandler Interface
Gives you more control over how customers or partners self-register for your Experience Cloud site by creating a class that implements Auth.ConfigurableSelfRegHandler. You choose the user information to collect, and how users identify themselves—with their email address, phone number, or another identifier. When verified, you create a customer or partner user and log in the user to your Experience Cloud site.

ConfirmUserRegistrationHandler Interface
Manages single sign-on (SSO) user mappings between Salesforce and a third-party identity provider. Use this interface to confirm user mappings before updating them.

ConnectedAppPlugin Class
Contains methods for extending the behavior of a connected app, for example, customizing how a connected app is invoked depending on the protocol used. This class gives you more control over the interaction between Salesforce and your connected app.

HeadlessSelfRegistrationHandler Interface
Creates customer and partner users during the Headless Registration Flow.

InvocationContext Enum
The context in which the connected app is invoked, such as the protocol flow used and the token type issued, if any. Developers can use the context information to write code that is unique to the type of invocation.

JWS Class
Contains methods that apply a digital signature to a JSON Web Token (JWT), using a JSON Web Signature (JWS) data structure. This class creates the signed JWT bearer token, which can be used to request an OAuth access token in the OAuth 2.0 JWT bearer token flow.

JWT Class
Generates the JSON Claims Set in a JSON Web Token (JWT). The resulting Base64-encoded payload can be passed as an argument to create an instance of the Auth.JWS class.

JWTBearerTokenExchange Class
Contains methods that POST the signed JWT bearer token to a token endpoint to request an access token, in the OAuth 2.0 JWT bearer token flow.

LightningLoginEligibility Enum
Contains a Lightning Login eligibility value used by the Auth.SessionManagement.getLightningLoginEligibility method.

LoginDiscoveryHandler Interface
Salesforce gives you the ability to log in users based on other verification methods than username and password. For example, it can prompt users to log in with their email, phone number, or another identifier like a Federation ID or device identifier. Login Discovery is available to these licenses: Customer Community, Customer Community Plus, External Identity, Partner Community, and Partner Community Plus.
LoginDiscoveryMethod Enum
Contains methods used to verify the user’s identity when the My Domain login process uses Login Discovery.

MyDomainLoginDiscoveryHandler Interface
The handler used to implement the My Domain Login Discovery page, which is an interview-based (two-step) login process. First the user is prompted for a unique identifier such as an email address or phone number. Then the handler determines (discovers) how to authenticate the user. Either the user enters a password or is directed to an identity provider’s login page.

OAuthRefreshResult Class
Stores the result of a AuthProviderPluginClass refresh method. OAuth authentication flow provides a refresh token that can be used to get a new access token. Access tokens have a limited lifetime as specified by the session timeout value. When an access token expires, use a refresh token to get a new access token.

RegistrationHandler Interface
Salesforce provides the ability to use an authentication provider, such as Facebook© or Janrain©, for single sign-on into Salesforce.

SamlJitHandler Interface
Use this interface to control and customize Just-in-Time user provisioning logic during SAML single sign-on.

SessionManagement Class
Contains methods for verifying users’ identity, creating custom login flows, customizing security levels, and defining trusted IP ranges for a current session.

SessionLevel Enum
An Auth.SessionLevel enum value is used by the SessionManagement.setSessionLevel method.

UserData Class
Stores user information for Auth.RegistrationHandler.

VerificationMethod Enum
Contains the different ways users can identify themselves when logging in. You can use it to implement mobile-centric passwordless login pages and to self-register (and deregister) verification methods.

VerificationPolicy Enum
The Auth.VerificationPolicy enum contains an identity verification policy value used by the SessionManagement.generateVerificationUrl method.

VerificationResult Class
Contains the result of a verification challenge that you invoke when you create your own Verify page. The challenge can be initiated by either the System.UserManagement.verifyPasswordlessLogin or System.UserManagement.verifySelfRegistration method.

Auth Exceptions
The Auth namespace contains some exception classes.

AuthConfiguration Class
Contains methods for configuring settings for users to log in to a Salesforce org using their authentication provider credentials instead of their Salesforce credentials. The authentication provider can be any authentication provider that supports the OpenID Connect protocol, such as Google, Facebook, or Twitter. Users log in to either an Experience Cloud site (https://MyDomainName.my.site.com) or your My Domain login URL (https://MyDomainName.my.salesforce.com).

Namespace
Auth
Example

This example shows how to call some methods on the Auth.AuthConfiguration class. Before you can run this sample, you must provide valid values for the URLs and developer name.

⚠️ Note: If you’re not using enhanced domains, your org’s My Domain URLs are different. For details, see My Domain URL Formats in Salesforce Help.

```java
String communityUrl = 'MyDomainName.my.site.com';
String startUrl = '<Add URL>,'
Auth.AuthConfiguration authConfig = new Auth.AuthConfiguration(communityUrl, startUrl);
List<AuthProvider> authPrvs = authConfig.getAuthProviders();
String bColor = authConfig.getBackgroundColor();
String fText = authConfig.getFooterText();

String sso = Auth.AuthConfiguration.getAuthProviderSsoUrl(communityUrl, startUrl, 'developerName');
```

AuthConfiguration Constructors

The following are constructors for AuthConfiguration.

⚠️ Note: The AuthConfiguration (networkId, startUrl) constructor is deprecated in API version 56.0 and later.

**AuthConfiguration(communityOrCustomUrl, startUrl)**

Creates an instance of the AuthConfiguration class using the specified URL for an Experience Cloud site or a My Domain subdomain and the start URL for authenticated users.

**Signature**

```java
public AuthConfiguration(String communityOrCustomUrl, String startUrl)
```

**Parameters**

- **communityOrCustomUrl**
  - Type: String
  - The URL for the domain, which can be a Salesforce subdomain created with My Domain (my.salesforce.com) or a subdomain of an Experience Cloud site (force.com).

- **startUrl**
  - Type: String
  - The page users see after successfully logging in to the Experience Cloud site or My Domain subdomain.

AuthConfiguration Methods

The following are methods for AuthConfiguration. Use these methods to manage and customize authentication for a Salesforce community.
getAllowInternalUserLoginEnabled()
Indicates whether the Experience Cloud site allows internal users to log in using the Experience Cloud site login page. To enable, admins configure the setting **Allow internal users to log in directly to the experience** on the Login & Registration page in Experience Workspaces. It's disabled by default.

getAuthConfig()
Returns the AuthConfig sObject, which represents the authentication options for an Experience Cloud site or Salesforce My Domain subdomain.

getAuthConfigProviders()
Returns the list of authentication providers configured for an Experience Cloud site or Salesforce My Domain subdomain.

getAuthProviderSsoDomainUrl(communityUrl, startUrl, developerName)
Returns the single sign-on URL for an Experience Cloud site subdomain.

getAuthProviderSsoUrl(communityUrl, startUrl, developerName)
Returns the single sign-on URL for an Experience Cloud site or Salesforce My Domain subdomain.

getBackgroundColor()
Returns the color for the background of the login page for a community.

getCertificateLoginEnabled(domainUrl)
Returns true if certificate-based authentication is enabled for the My Domain URL.

getCertificateLoginUrl(domainUrl, startUrl)
Returns the certificate-based authentication endpoint for the My Domain URL if the org has certificate-based authentication enabled.

defaultProfileForRegistration()
Returns the profile ID assigned to new community users.

getFooterText()
Returns the text at the bottom of the login page for a community.

getForgotPasswordUrl()
Returns the URL for the standard or custom Forgot Password page that is specified for an Experience Cloud site or portal by the administrator.

getHeadlessForgotPasswordEnabled()
Returns true if the Headless Forgot Password Flow is enabled.

getHeadlessFrgtPswEnabled()
This method will be deprecated in a future release. Use the getHeadlessForgotPasswordEnabled() method in this class instead.

getHeadlessPasswordlessLoginEnabled()
Determines if headless passwordless login is enabled.

getHeadlessRegistrationEnabled()
Determines if the Headless Registration Flow is enabled.

getLogoUrl()
Returns the location of the icon image at the bottom of the login page for a community.
getRightFrameUrl()
Returns the URL for the right-frame content to display on the right side of the Experience Cloud site login page. The admin supplies the URL.

getSamlProviders()
Returns the list of SAML-based authentication providers available for an Experience Cloud site or Salesforce My Domain subdomain.

g getSamlSsoUrl(communityUrl, startURL, samlId)
Returns the single sign-on URL for an Experience Cloud site or Salesforce My Domain subdomain.

getSelfRegistrationEnabled()
Indicates whether the current community allows new users to create their own account by filling out a registration form.

getSelfRegistrationUrl()
Returns the location of the self-registration page for new users to sign up for an account with a community.

getStartUrl()
Returns the start page of an Experience Cloud site or Salesforce My Domain subdomain. This URL is the first page that users see when they log in.

g getUsernamePasswordEnabled()
Indicates whether the current community is set to display a login form asking for a username and password. You can configure the community not to request a username and password if it is for unauthenticated users or users logging in with a third-party authentication provider.

isCommunityUsingSiteAsContainer()
Returns true if the Experience Cloud site uses Site.com pages; otherwise, returns false.

goingalAllowInternalUserLoginEnabled()
Indicates whether the Experience Cloud site allows internal users to log in using the Experience Cloud site login page. To enable, admins configure the setting Allow internal users to log in directly to the experience on the Login & Registration page in Experience Workspaces. It's disabled by default.

Signature
public Boolean getAllowInternalUserLoginEnabled()

Return Value
Type: Boolean

Usage
If true, internal users log in to an Experience Cloud site from the site's login page with their internal credentials. If they navigate to their internal org from the Experience Cloud site, they don't have to log in again.

goingalAuthConfig()
Returns the AuthConfig sObject, which represents the authentication options for an Experience Cloud site or Salesforce My Domain subdomain.
Signature

```java
public AuthConfig getAuthConfig()
```

Return Value

Type: AuthConfig

The AuthConfig sObject for the Experience Cloud site or Salesforce My Domain subdomain.

**getAuthConfigProviders()**

Returns the list of authentication providers configured for an Experience Cloud site or Salesforce My Domain subdomain.

Signature

```java
public List<AuthConfigProviders> getAuthConfigProviders()
```

Return Value

Type: List<AuthConfigProviders>

A list of authentication providers (AuthConfigProviders sObjects), which are children of the AuthProvider sObject.

**getAuthProviders()**

Returns the list of authentication providers available for an Experience Cloud site or Salesforce My Domain subdomain.

Signature

```java
public List<AuthProvider> getAuthProviders()
```

Return Value

Type: List<AuthProvider>

A list of authentication providers (AuthProvider sObjects) for the Experience Cloud site or My Domain subdomain.

**getAuthProviderSsoDomainUrl(communityUrl, startUrl, developerName)**

Returns the single sign-on URL for an Experience Cloud site subdomain.

⚠️ **Note:** For better performance, we recommend using this method instead of `getAuthProviderSsoUrl`. If the authentication provider has `User Subdomain for Callback` enabled, changing the single sign-on URL also changes the callback URL to use the Experience Cloud site subdomain. Before switching to this method, update the callback URL in your third-party applications to avoid getting an invalid callback URL error during single sign-on.

Signature

```java
public static String getAuthProviderSsoDomainUrl(String communityUrl, String startUrl, String developerName)
```
Parameters

communityUrl
Type: String
The URL for the Experience Cloud site subdomain. If null or specified as an empty string, you get the single sign-on URL for the org’s My Domain.

startUrl
Type: String
The page that users see after logging in to the Experience Cloud site subdomain.

developerName
Type: String
The unique name of the authentication provider.

Return Value
Type: String
The Single Sign-On Initialization URL for the Experience Cloud site subdomain.

getAuthProviderSsoUrl(communityUrl, startUrl, developerName)
Returns the single sign-on URL for an Experience Cloud site or Salesforce My Domain subdomain.

Signature

public static String getAuthProviderSsoUrl(String communityUrl, String startUrl, String developerName)

Parameters

communityUrl
Type: String
The URL for the Experience Cloud site or My Domain subdomain. If not null and not specified as an empty string, you get the URL for the Experience Cloud site. If null or specified as an empty string, you get the URL for a custom domain.

startUrl
Type: String
The page that users see after logging in to the Experience Cloud site or My Domain subdomain.

developerName
Type: String
The unique name of the authentication provider.

Return Value
Type: String
The Single Sign-On Initialization URL for the Experience Cloud site or Salesforce My Domain subdomain.
getBackgroundColor()
Returns the color for the background of the login page for a community.

Signature
public String getBackgroundColor()

Return Value
Type: String

gCertificateLoginEnabled(domainUrl)
Returns true if certificate-based authentication is enabled for the My Domain URL.

Signature
public Boolean getCertificateLoginEnabled(String domainUrl)

Parameters
domainUrl
Type: String
The My Domain URL that is being checked for certificate-based authentication.

Return Value
Type: Boolean

gCertificateLoginUrl(domainUrl, startUrl)
Returns the certificate-based authentication endpoint for the My Domain URL if the org has certificate-based authentication enabled.

Signature
public static String getCertificateLoginUrl(String domainUrl, String startUrl)

Parameters
domainUrl
Type: String
The My Domain URL being checked for its certificate-based authentication endpoint.

startUrl
Type: String
The page that the user is directed to after logging in to the My Domain with certificate-based authentication.

Return Value
Type: String
The certificate-based authentication endpoint for the My Domain URL:
mydomainURL:8443/services/certauth?startURL=startURLParam

**getDefaultProfileForRegistration()**
Returns the profile ID assigned to new community users.

**Signature**

```java
public String getDefaultProfileForRegistration()
```

**Return Value**

Type: `String`
The profile ID.

**getFooterText()**
Returns the text at the bottom of the login page for a community.

**Signature**

```java
public String getFooterText()
```

**Return Value**

Type: `String`
The text string displayed at the bottom of the login page, for example “Log in with an existing account.”

**getForgotPasswordUrl()**
Returns the URL for the standard or custom Forgot Password page that is specified for an Experience Cloud site or portal by the administrator.

**Signature**

```java
public String getForgotPasswordUrl()
```

**Return Value**

Type: `String`
URL for the standard or custom Forgot Password page.

**getHeadlessForgotPasswordEnabled()**
Returns true if the Headless Forgot Password Flow is enabled.

**Signature**

```java
public Boolean getHeadlessForgotPasswordEnabled()
```
Return Value
Type: Boolean

getHeadlessFrgtPswEnabled()
This method will be deprecated in a future release. Use the getHeadlessForgotPasswordEnabled() method in this class instead.

Signature
public Boolean getHeadlessFrgtPswEnabled()

Return Value
Type: Boolean

getHeadlessPasswordlessLoginEnabled()
Determines if headless passwordless login is enabled.

Signature
public Boolean getHeadlessPasswordlessLoginEnabled()

Return Value
Type: Boolean
Returns true if headless passwordless login is enabled.

getHeadlessRegistrationEnabled()
Determines if the Headless Registration Flow is enabled.

Signature
public Boolean getHeadlessRegistrationEnabled()

Return Value
Type: Boolean
Returns true if headless registration is enabled.

getLogoUrl()
Returns the location of the icon image at the bottom of the login page for a community.

Signature
public String getLogoUrl()
Return Value
Type: String
The path to the icon image.

**getRightFrameUrl()**
Returns the URL for the right-frame content to display on the right side of the Experience Cloud site login page. The admin supplies the URL.

**Signature**
```java
public String getLoginRightFrameUrl()
```

**Return Value**
Type: String
URL for the right-frame content of the Experience Cloud site login page. Salesforce creates an inline (iframe) on the right side of the login page to display the contents specified by the URL.

**getSamlProviders()**
Returns the list of SAML-based authentication providers available for an Experience Cloud site or Salesforce My Domain subdomain.

**Signature**
```java
public List<SamlSsoConfig> getSamlProviders()
```

**Return Value**
Type: List<SamlSsoConfig>
A list of SAML-based authentication providers, which are SamlSsoConfig sObjects.

**getSamlSsoUrl(communityUrl, startURL, samlId)**
Returns the single sign-on URL for an Experience Cloud site or Salesforce My Domain subdomain.

**Signature**
```java
public static String getSamlSsoUrl(String communityUrl, String startURL, String samlId)
```

**Parameters**
- **communityUrl**
  Type: String
  The URL for the Experience Cloud site or My Domain subdomain. If not null and not specified as an empty string, you get the URL for the Experience Cloud site. If null or specified as an empty string, you get the URL for a My Domain subdomain.
- **startUrl**
  Type: String
The page users see after successfully logging in to the Experience Cloud site or My Domain subdomain

samlId
Type: String
The unique identifier of the SamlSsoConfig standard object for the Experience Cloud site or My Domain subdomain

Return Value
Type: String
The Single Sign-On Initialization URL for the Experience Cloud site or Salesforce My Domain subdomain.

getSelfRegistrationEnabled() 
Indicates whether the current community allows new users to create their own account by filling out a registration form.

Signature
public Boolean getSelfRegistrationEnabled()

Return Value
Type: Boolean

getSelfRegistrationUrl() 
Returns the location of the self-registration page for new users to sign up for an account with a community.

Signature
public String getSelfRegistrationUrl()

Return Value
Type: String
The location of the self-registration page.

getStartUrl() 
Returns the start page of an Experience Cloud site or Salesforce My Domain subdomain. This URL is the first page that users see when they log in.

Signature
public String getStartUrl()

Return Value
Type: String
The location of the start page for the Experience Cloud site or My Domain subdomain.
getUsernamePasswordEnabled()

Indicates whether the current community is set to display a login form asking for a username and password. You can configure the community not to request a username and password if it is for unauthenticated users or users logging in with a third-party authentication provider.

Signature

public Boolean getUsernamePasswordEnabled()

Return Value

Type: Boolean

isCommunityUsingSiteAsContainer()

Returns true if the Experience Cloud site uses Site.com pages; otherwise, returns false.

Signature

public Boolean isCommunityUsingSiteAsContainer()

Return Value

Type: Boolean

AuthProviderCallbackState Class

Provides request HTTP headers, body, and query parameters to the AuthProviderPlugin.handleCallback method for user authentication. This class allows you to group the information passed in rather than passing headers, body, and query parameters individually.

Namespace

Auth

IN THIS SECTION:

AuthProviderCallbackState Constructors

AuthProviderCallbackState Properties

SEE ALSO:

handleCallback(authProviderConfiguration, callbackState)

AuthProviderCallbackState Constructors

The following are constructors for AuthProviderCallbackState.
AuthProviderCallbackState class using the specified HTTP headers, body, and query parameters of the authentication request.

**AuthProviderCallbackState**

Creates an instance of the `AuthProviderCallbackState` class using the specified HTTP headers, body, and query parameters of the authentication request.

**Signature**

```java
public AuthProviderCallbackState(Map<String,String> headers, String body, Map<String,String> queryParameters)
```

**Parameters**

- `headers`
  Type: `Map<String,String>`
  The HTTP headers of the authentication request.

- `body`
  Type: `String`
  The HTTP body of the authentication request.

- `queryParameters`
  Type: `Map<String,String>`
  The HTTP query parameters of the authentication request.

**AuthProviderCallbackState Properties**

The following are properties for `AuthProviderCallbackState`.

**IN THIS SECTION:**

- `body`
  The HTTP body of the authentication request.
- `headers`
  The HTTP headers of the authentication request.
- `queryParameters`
  The HTTP query parameters of the authentication request.

**body**

The HTTP body of the authentication request.

**Signature**

```java
public String body {get; set;}
```
Property Value
Type: String

headers
The HTTP headers of the authentication request.

Signature
public Map<String,String> headers {get; set;}

Property Value
Type: Map<String,String>

queryParameters
The HTTP query parameters of the authentication request.

Signature
public Map<String,String> queryParameters {get; set;}

AuthProviderPlugin Interface
This interface is deprecated. For new development, use the abstract class Auth.AuthProviderPluginClass to create a custom OAuth-based authentication provider plug-in for single sign-on into Salesforce.

Namespace
Auth

Usage
Deprecated. Existing implementations that use Auth.AuthProviderPlugin still work. For new development, use Auth.AuthProviderPluginClass.

IN THIS SECTION:
 AuthProviderPlugin Methods
 AuthProviderPlugin Example Implementation
AuthProviderPlugin Methods

The following methods are for AuthProviderPlugin, which, as of API version 39.0, is deprecated. Use the methods in AuthProviderPluginClass instead.

IN THIS SECTION:

- `getCustomMetadataType()`
  Deprecation as of API version 39.0. Use the corresponding method in Auth.AuthProviderPluginClass.

- `getUserInfo(authProviderConfiguration, response)`
  Deprecation as of API version 39.0. Use the corresponding method in Auth.AuthProviderPluginClass.

- `handleCallback(authProviderConfiguration, callbackState)`
  Deprecation as of API version 39.0. Use the corresponding method in Auth.AuthProviderPluginClass.

- `initiate(authProviderConfiguration, stateToPropagate)`
  Deprecation as of API version 39.0. Use the corresponding method in Auth.AuthProviderPluginClass.

SEE ALSO:

- Salesforce Help: Create a Custom External Authentication Provider

`getCustomMetadataType()`

Deprecation as of API version 39.0. Use the corresponding method in Auth.AuthProviderPluginClass.

**Signature**

```java
public String getCustomMetadataType()
```

**Return Value**

*Type: String*

The custom metadata type API name for the authentication provider.

**Usage**

Returns the custom metadata type API name for a custom OAuth-based authentication provider for single sign-on to Salesforce. The `getCustomMetadataType()` method returns only custom metadata type names. It does not return custom metadata record names.

`getUserInfo(authProviderConfiguration, response)`

Deprecation as of API version 39.0. Use the corresponding method in Auth.AuthProviderPluginClass.

**Signature**

```java
public Auth.UserData getUserInfo(Map<String,String> authProviderConfiguration, Auth.AuthProviderTokenResponse response)
```
Parameters

authProviderConfiguration
Type: Map<String,String>
The configuration for the custom authentication provider. When you create a custom metadata type in Salesforce, the configuration populates with the custom metadata type default values. Or you can set the configuration with values you enter when you create the custom provider in Auth. Providers in Setup.

response
Type: Auth.AuthProviderTokenResponse
The OAuth access token, OAuth secret or refresh token, and state provided by the authentication provider to authenticate the current user.

Return Value

Type: Auth.UserData
Creates a new instance of the Auth.UserData class.

Usage

Returns information from the custom authentication provider about the current user. The registration handler and other authentication provider flows use this information.

handleCallback(authProviderConfiguration, callbackState)
Deprecated as of API version 39.0. Use the corresponding method in Auth.AuthProviderPluginClass.

Signature

public Auth.AuthProviderTokenResponse handleCallback(Map<String,String> authProviderConfiguration, Auth.AuthProviderCallbackState callbackState)

Parameters

authProviderConfiguration
Type: Map<String,String>
The configuration for the custom authentication provider. When you create a custom metadata type in Salesforce, the configuration populates with the custom metadata type default values. Or you can set the configuration with values you enter when you create the custom provider in Auth. Providers in Setup.

callbackState
Type: Auth.AuthProviderCallbackState
The class that contains the HTTP headers, body, and queryParams of the authentication request.

Return Value

Type: Auth.AuthProviderTokenResponse
Creates an instance of the AuthProviderTokenResponse class.
Usage
Uses the authentication provider’s supported authentication protocol to return an OAuth access token, OAuth secret or refresh token, and the state passed in when the request for the current user was initiated.

**initiate(authProviderConfiguration, stateToPropagate)**
Deprecated as of API version 39.0. Use the corresponding method in Auth.AuthProviderPluginClass.

**Signature**

```java
public System.PageReference initiate(Map<String,String> authProviderConfiguration,
 String stateToPropagate)
```

**Parameters**

- **authProviderConfiguration**
  Type: `Map<String,String>`
  The configuration for the custom authentication provider. When you create a custom metadata type in Salesforce, the configuration populates with the custom metadata type default values. Or you can set the configuration with values you enter when you create the custom provider in Auth. Providers in Setup.

- **stateToPropagate**
  Type: `String`
  The state passed in to initiate the authentication request for the user.

**Return Value**

Type: `System.PageReference`

The URL of the page where the user is redirected for authentication.

Usage
Returns the URL where the user is redirected for authentication.

**AuthProviderPlugin Example Implementation**
We’ve removed the example implementation for the `Auth.AuthProviderPlugin` interface because we’ve deprecated the interface and replaced it with an abstract class. See `AuthProviderPluginClass` Class.

**AuthProviderPluginClass Class**
Contains methods to create a custom OAuth-based authentication provider plug-in for single sign-on in to Salesforce. Use this class to create a custom authentication provider plug-in if you can’t use one of the authentication providers that Salesforce provides.

**Namespace**

`Auth`
Usage

To create a custom authentication provider for single sign-on, create a class that extends Auth.AuthProviderPluginClass. This class allows you to store the custom configuration for your authentication provider and handle authentication protocols when users log in to Salesforce with their login credentials for an external service provider. In Salesforce, the class that implements this interface appears in the Provider Type drop-down list in Auth. Providers in Setup. Make sure that the user you specify to run the class has “Customize Application” and “Manage Auth. Providers” permissions.

As of API version 39.0, use the abstract class AuthProviderPluginClass to create a custom external authentication provider. This class replaces the AuthProviderPlugin interface. If you've already implemented a custom authentication provider plug-in using the interface, it still works. However, use AuthProviderPluginClass to extend your plug-in. If you haven't created an interface, create a custom authentication provider plug-in by extending this abstract class. For more information, see AuthProviderPluginClass Code Example.

IN THIS SECTION:

AuthProviderPluginClass Methods
AuthProviderPluginClass Code Example

AuthProviderPluginClass Methods

The AuthProviderPluginClass methods don't support DML options.

IN THIS SECTION:

getCustomMetadataType()
Returns the custom metadata type API name for a custom OAuth-based authentication provider for single sign-on to Salesforce.

getUserInfo(authProviderConfiguration, response)
Returns information from the custom authentication provider about the current user. This information is used by the registration handler and in other authentication provider flows.

handleCallback(authProviderConfiguration, callbackState)
Uses the authentication provider’s supported authentication protocol to return an OAuth access token, OAuth secret or refresh token, and the state passed in when the request for the current user was initiated.

initiate(authProviderConfiguration, stateToPropagate)
Returns the URL where the user is redirected for authentication.

refresh(authProviderConfiguration, refreshToken)
Returns a new access token, which is used to update an expired access token.

getCustomMetadataType()

Returns the custom metadata type API name for a custom OAuth-based authentication provider for single sign-on to Salesforce.

Signature

public String getCustomMetadataType()
Return Value

Type: String

The custom metadata type API name for the authentication provider.

Usage

The `getCustomMetadataType()` method returns only custom metadata type names. It does not return custom metadata record names. As of API version 39.0, use this method when extending `AuthProviderPluginClass` to create a custom external authentication provider.

`getUserInfo(authProviderConfiguration, response)`

Returns information from the custom authentication provider about the current user. This information is used by the registration handler and in other authentication provider flows.

Signature

```java
public Auth.UserData getUserInfo(Map<String,String> authProviderConfiguration, Auth.AuthProviderTokenResponse response)
```

Parameters

`authProviderConfiguration`

Type: `Map<String,String>`

The configuration for the custom authentication provider. When you create a custom metadata type in Salesforce, the configuration populates it with the custom metadata type default values. Or you can set the configuration with values that you enter when you create the custom provider in Auth. Providers in Setup.

`response`

Type: `Auth.AuthProviderTokenResponse`

The OAuth access token, OAuth secret or refresh token, and state provided by the authentication provider to authenticate the current user.

Return Value

Type: `Auth.UserData`

Creates a new instance of the `Auth.UserData` class.

Usage

As of API version 39.0, use this method when extending `AuthProviderPluginClass` to create a custom authentication provider.

Note: You might choose to get user information in the response from the `handleCallback` method or by another method. However, you must still call `getUserInfo` in the custom authentication handler to avoid getting errors about mixing objects. For example, if you don’t call `getUserInfo`, and then try to insert a contact in the `Auth.RegistrationHandler.createUser` method, you get the error, “You cannot mix EntityObjects with different UddInfos within one transaction.”
To avoid this error, call `getUserInfo` with dummy user information as follows.

```java
HttpRequest req = new HttpRequest();
String url = 'https://login.salesforce.com/';
req.setEndpoint(url);
req.setMethod('GET');
Http http = new Http();
HTTPResponse res = http.send(req);
```

**handleCallback** *(authProviderConfiguration, callbackState)*

Uses the authentication provider’s supported authentication protocol to return an OAuth access token, OAuth secret or refresh token, and the state passed in when the request for the current user was initiated.

**Signature**

```java
public Auth.AuthProviderTokenResponse handleCallback(Map<String,String> authProviderConfiguration, Auth.AuthProviderCallbackState callbackState)
```

**Parameters**

- **authProviderConfiguration**
  Type: `Map<String,String>`
  The configuration for the custom authentication provider. When you create a custom metadata type in Salesforce, the configuration populates with the custom metadata type default values. Or you can set the configuration with values you enter when you create the custom provider in Auth. Providers in Setup.

- **callbackState**
  Type: `Auth.AuthProviderCallbackState`
  The class that contains the HTTP headers, body, and queryParams of the authentication request.

**Return Value**

Type: `Auth.AuthProviderTokenResponse`

Creates an instance of the `AuthProviderTokenResponse` class.

**Usage**

As of API version 39.0, use this method when extending `Auth.AuthProviderPluginClass` to create a custom authentication provider.

**initiate** *(authProviderConfiguration, stateToPropagate)*

Returns the URL where the user is redirected for authentication.

**Signature**

```java
public System.PageReference initiate(Map<String,String> authProviderConfiguration, String stateToPropagate)
```
Parameters

authProviderConfiguration
Type: Map<String,String>
The configuration for the custom authentication provider. When you create a custom metadata type in Salesforce, the configuration populates with the custom metadata type default values. Or you can set the configuration with values you enter when you create the custom provider in Auth. Providers in Setup.

stateToPropagate
Type: String
The state passed in to initiate the authentication request for the user.

Return Value

Type: System.PageReference
The URL of the page where the user is redirected for authentication.

Usage

As of API version 39.0, use this method when extending Auth.AuthProviderPluginClass to create a custom authentication provider.

refresh(authProviderConfiguration, refreshToken)

Returns a new access token, which is used to update an expired access token.

Signature

public Auth.OAuthRefreshResult refresh(Map<String,String> authProviderConfiguration, String refreshToken)

Parameters

authProviderConfiguration
Type: Map<String,String>
The configuration for the custom authentication provider. When you create a custom metadata type in Salesforce, the configuration populates with the custom metadata type default values. Or you can set the configuration with values you enter when you create the custom provider in Auth. Providers in Setup.

refreshToken
Type: String
The refresh token for the user who is logged in.

Return Value

Type: Auth.OAuthRefreshResult
Returns the new access token, or an error message if an error occurs.
Usage

A successful request returns a `Auth.OAuthRefreshResult` with the access token and refresh token in the response. If you receive an error, make sure that you set the error string to the error message. A NULL error string indicates no error.

The refresh method works only with named credentials; it doesn’t respect the standard OAuth refresh flow. The refresh method with named credentials works only if the earlier request returns a 401.

AuthProviderPluginClass Code Example

The following example demonstrates how to implement a custom Auth. provider plug-in using the abstract class, `Auth.AuthProviderPluginClass`.

```java
global class Concur extends Auth.AuthProviderPluginClass {

    // Use this URL for the endpoint that the
    // authentication provider calls back to for configuration.
    public String redirectUrl;
    private String key;
    private String secret;

    // Application redirection to the Concur website for
    // authentication and authorization.
    private String authUrl;

    // URI to get the new access token from concur using the GET verb.
    private String accessTokenUrl;

    // Api name for the custom metadata type created for this auth provider.
    private String customMetadataTypeApiName;

    // Api URL to access the user in Concur
    private String userAPIUrl;

    // Version of the user api URL to access data from Concur
    private String userAPIVersionUrl;

    global String getCustomMetadataType() {
        return customMetadataTypeApiName;
    }

    global PageReference initiate(Map<string,string> authProviderConfiguration, String stateToPropagate) {
        authUrl = authProviderConfiguration.get('Auth_Url__c');
        key = authProviderConfiguration.get('Key__c');

        // Here the developer can build up a request of some sort.
        // Ultimately, they return a URL where we will redirect the user.
        String url = authUrl + '?client_id='+ key +'
+&scope=USER,EXPRPT,LIST&redirect_uri=' + redirectUrl + '&state=' + stateToPropagate;
        return new PageReference(url);
    }
}
```
global Auth.AuthProviderTokenResponse handleCallback(Map<string,string> authProviderConfiguration, Auth.AuthProviderCallbackState state)
{
    // Here, the developer will get the callback with actual protocol.
    // Their responsibility is to return a new object called
    // AuthProviderTokenResponse.
    // This will contain an optional accessToken and refreshToken
    key = authProviderConfiguration.get('Key__c');
    secret = authProviderConfiguration.get('Secret__c');
    accessTokenUrl = authProviderConfiguration.get('Access_Token_Url__c');

    Map<String,String> queryParams = state.queryParameters;
    String code = queryParams.get('code');
    String sfdcState = queryParams.get('state');

    HttpRequest req = new HttpRequest();
    String url = accessTokenUrl+'?code=' + code + '&client_id=' + key +
    '&client_secret=' + secret;
    req.setEndpoint(url);
    req.setHeader('Content-Type','application/xml');
    req.setMethod('GET');
    Http http = new Http();
    HTTPResponse res = http.send(req);
    String responseBody = res.getBody();
    String token = getTokenValueFromResponse(responseBody, 'Token', null);
    return new Auth.AuthProviderTokenResponse('Concur', token,
    'refreshToken', sfdcState);
}

global Auth.UserData getUserInfo(Map<string,string> authProviderConfiguration,
    Auth.AuthProviderTokenResponse response)
{
    //Here the developer is responsible for constructing an
    //Auth.UserData object
    String token = response.oauthToken;
    HttpRequest req = new HttpRequest();
    userAPIUrl = authProviderConfiguration.get('API_User_Url__c');
    userAPIVersionUrl = authProviderConfiguration.get
    ('API_User_Version_Url__c');
    req.setHeader('Authorization', 'OAuth ' + token);
    req.setEndpoint(userAPIUrl);
    req.setHeader('Content-Type','application/xml');
    req.setMethod('GET');
    Http http = new Http();
    HTTPResponse res = http.send(req);
    String responseBody = res.getBody();
    String id = getTokenValueFromResponse(responseBody, 'LoginId',userAPIVersionUrl);
    String fname = getTokenValueFromResponse(responseBody, 'FirstName', userAPIVersionUrl);
}
String lname = getTokenValueFromResponse(responseBody, 'LastName', userAPIVersionUrl);
String fname = fname + ' ' + lname;
String uname = getTokenValueFromResponse(responseBody, 'EmailAddress', userAPIVersionUrl);
String locale = getTokenValueFromResponse(responseBody, 'LocaleName', userAPIVersionUrl);
Map<String,String> provMap = new Map<String,String>(
    'what1', 'noidea1',
    'what2', 'noidea2'
);
return new Auth.UserData(id, fname, lname, flname, uname, 'what', locale, null, 'Concur', null, provMap);

private String getTokenValueFromResponse(String response, String token, String ns) {
    docx.load(response);
    String ret = null;
    dom.XmlNode xroot = docx.getrootelement();
    if(xroot != null){ ret = xroot.getChildElement(token, ns).getText();
    }
    return ret;
}

Sample Test Classes
The following example contains test classes for the Concur class.

@IsTest
public class ConcurTestClass {
    private static final String OAUTH_TOKEN = 'testToken';
    private static final String STATE = 'mocktestState';
    private static final String REFRESH_TOKEN = 'refreshToken';
    private static final String LOGIN_ID = 'testLoginId';
    private static final String USERNAME = 'testUsername';
    private static final String FIRST_NAME = 'testFirstName';
    private static final String LAST_NAME = 'testLastName';
    private static final String EMAIL_ADDRESS = 'testEmailAddress';
    private static final String LOCALE_NAME = 'testLocalName';
    private static final String FULL_NAME = FIRST_NAME + ' ' + LAST_NAME;
    private static final String PROVIDER = 'Concur';
    private static final String REDIRECT_URL = 'http://localhost/services/authcallback/orgId/Concur';
    private static final String KEY = 'testKey';
    private static final String SECRET = 'testSecret';
    private static final String STATE_TO_PROPAGATE = 'testState';
    private static final String ACCESS_TOKEN_URL =
private static final String API_USER_VERSION_URL = 'http://www.dummyhost.com/user/20/1';
private static final String AUTH_URL = 'http://www.dummy.com/authurl';
private static final String API_USER_URL = 'www.concursolutions.com/user/api';

// In the real world scenario, the key and value would be read
// from the (custom fields in) custom metadata type record.
private static Map<String, String> setupAuthProviderConfig ()
{
    Map<String, String> authProviderConfiguration = new Map<String, String>();
    authProviderConfiguration.put('Key__c', KEY);
    authProviderConfiguration.put('Auth_Url__c', AUTH_URL);
    authProviderConfiguration.put('Secret__c', SECRET);
    authProviderConfiguration.put('Access_Token_Url__c', ACCESS_TOKEN_URL);
    authProviderConfiguration.put('API_User_Url__c', API_USER_URL);
    authProviderConfiguration.put('API_User_Url__c', API_USER_URL);
    authProviderConfiguration.put('API_User_Url__c', API_USER_URL);
    authProviderConfiguration.put('Redirect_Url__c', REDIRECT_URL);
    return authProviderConfiguration;
}

dynamic testMethod void testInitiateMethod()
{
    String stateToPropogate = 'mocktestState';
    Map<String, String> authProviderConfiguration = setupAuthProviderConfig();
    Concur concurCls = new Concur();
    concurCls.redirectUrl = authProviderConfiguration.get('Redirect_Url__c');

    PageReference expectedUrl = new PageReference(
        authProviderConfiguration.get('Auth_Url__c') + '?client_id=' +
        authProviderConfiguration.get('Key__c') + '&scope=USER,EXPRPT,LIST&redirect_uri=' +
        authProviderConfiguration.get('Redirect_Url__c') + '&state=' +
        STATE_TO_PROPOGATE);
    PageReference actualUrl = concurCls.initiate(authProviderConfiguration, STATE_TO_PROPOGATE);
    System.assertEquals(expectedUrl.getUrl(), actualUrl.getUrl());
}

dynamic testMethod void testHandleCallback()
{
    Map<String, String> authProviderConfiguration = setupAuthProviderConfig();
    Concur concurCls = new Concur();
    concurCls.redirectUrl = authProviderConfiguration.get('Redirect_Url__c');

    Test.setMock(HttpCalloutMock.class, new
}
ConcurMockHttpResponseGenerator();

Map<String,String> queryParams = new Map<String,String>();
queryParams.put('code', 'code');
queryParams.put('state', authProviderConfiguration.get('State_c'));
Auth.AuthProviderCallbackState cbState =
new Auth.AuthProviderCallbackState(null, null, queryParams);
Auth.AuthProviderTokenResponse actualAuthProvResponse =
concurrCls.handleCallback(authProviderConfiguration, cbState);
Auth.AuthProviderTokenResponse expectedAuthProvResponse =
new Auth.AuthProviderTokenResponse(
'Concur', OAUTH_TOKEN, REFRESH_TOKEN, null);

System.assertEquals(expectedAuthProvResponse.provider,
actualAuthProvResponse.provider);
System.assertEquals(expectedAuthProvResponse.oauthToken,
actualAuthProvResponse.oauthToken);
System.assertEquals(expectedAuthProvResponse.oauthSecretOrRefreshToken,
actualAuthProvResponse.oauthSecretOrRefreshToken);
System.assertEquals(expectedAuthProvResponse.state,
actualAuthProvResponse.state);

} 

static testMethod void testGetUserInfo()
{
    Map<String,String> authProviderConfiguration =
    setupAuthProviderConfig();
    Concur concurCls = new Concur();
    Test.setMock(HttpCalloutMock.class, new
ConcurMockHttpResponseGenerator());

    Auth.AuthProviderTokenResponse response =
new Auth.AuthProviderTokenResponse(
 PROVIDER, OAUTH_TOKEN, 'sampleOauthSecret', STATE);
    Auth.UserData actualUserData = concurCls.getUserInfo(
authProviderConfiguration, response) ;

    Map<String,String> provMap = new Map<String,String>();
provMap.put('key1', 'value1');
provMap.put('key2', 'value2');

    Auth.UserData expectedUserData = new Auth.UserData(LOGIN_ID,
FIRST_NAME, LAST_NAME, FULL_NAME, EMAIL_ADDRESS,
null, LOCALE_NAME, null, PROVIDER, null, provMap);

    System.assertNotEquals(expectedUserData,null);
    System.assertEquals(expectedUserData.firstName,
actualUserData.firstName);
    System.assertEquals(expectedUserData.lastName,
actualUserData.lastName);
    System.assertEquals(expectedUserData.fullName,
actualUserData.fullName);
}
AuthProviderTokenResponse Class

Stores the response from theAuthProviderPlugin.handleCallback method.
AuthProviderTokenResponse Class

Namespace
Auth

IN THIS SECTION:
AuthProviderTokenResponse Constructors
AuthProviderTokenResponse Properties

AuthProviderTokenResponse Constructors
The following are constructors for AuthProviderTokenResponse.

IN THIS SECTION:
AuthProviderTokenResponse(provider, oauthToken, oauthSecretOrRefreshToken, state)

AuthProviderTokenResponse(provider, oauthToken, oauthSecretOrRefreshToken, state)

Creates an instance of the AuthProviderTokenResponse class using the specified authentication provider, OAuth access token, OAuth secret or refresh token, and state for a custom authentication provider plug-in.

AuthProviderTokenResponse(provider, oauthToken, oauthSecretOrRefreshToken, state)

Creates an instance of the AuthProviderTokenResponse class using the specified authentication provider, OAuth access token, OAuth secret or refresh token, and state for a custom authentication provider plug-in.

Signature

public AuthProviderTokenResponse(String provider, String oauthToken, String oauthSecretOrRefreshToken, String state)

Parameters

provider
Type: String
The custom authentication provider.

oauthToken
Type: String
The OAuth access token.

oauthSecretOrRefreshToken
Type: String
The OAuth secret or refresh token for the currently logged-in user.

state
Type: String
The state passed in to initiate the authentication request for the user.
AuthProviderTokenResponse Properties

The following are properties for AuthProviderTokenResponse.

IN THIS SECTION:

oauthSecretOrRefreshToken
The OAuth secret or refresh token for the currently logged-in user.

oauthToken
The OAuth access token.

provider
The authentication provider.

state
The state passed in to initiate the authentication request for the user.

oauthSecretOrRefreshToken
The OAuth secret or refresh token for the currently logged-in user.

Signature

public String oauthSecretOrRefreshToken {get; set;}

Property Value
Type: String

oauthToken
The OAuth access token.

Signature

public String oauthToken {get; set;}

Property Value
Type: String

provider
The authentication provider.

Signature

public String provider {get; set;}


Property Value
Type: String

state
The state passed in to initiate the authentication request for the user.

Signature
public String state {get; set;}

Property Value
Type: String

AuthToken Class
Contains methods for providing the access token associated with an authentication provider for an authenticated user, except for the Janrain provider.

Namespace
Auth

AuthToken Methods
The following are methods for AuthToken. All methods are static.

IN THIS SECTION:

getAccessToken(authProviderId, providerName)
Returns an access token for the current user using the specified 18-character identifier of anAuthProvider definition in your org and the proper name of the third party, such as Salesforce or Facebook. Note that querying the ProviderType field on the AuthProvider object sometimes returns a value that differs from the expected provider name value. For example, for Open ID Connect providers, OpenIdConnect is the ProviderType value for the AuthProvider object, but the expected providerName is Open ID Connect.

getAccessTokenMap(authProviderId, providerName)
Returns a map from the third-party identifier to the access token for the currently logged-in Salesforce user. The identifier value depends on the third party. For example, for Salesforce it would be the user ID, while for Facebook it would be the user number. Note that querying the ProviderType field on the AuthProvider object sometimes returns a value that differs from the expected provider name value. For example, for Open ID Connect providers, OpenIdConnect is the ProviderType value for the AuthProvider object, but the expected providerName is Open ID Connect.

refreshAccessToken(authProviderId, providerName, oldAccessToken)
Returns a map from the third-party identifier containing a refreshed access token for the currently logged-in Salesforce user. Note that querying the ProviderType field on the AuthProvider object sometimes returns a value that differs from the expected provider name value. For example, for Open ID Connect providers, OpenIdConnect is the ProviderType value for the AuthProvider object, but the expected providerName is Open ID Connect.
revokeAccess(authProviderId, providerName, userId, remoteIdentifier)

Revoke the access token for a specified social sign-on user from a third-party service such as Facebook©. You can use this method only if the IsNotSsoUsable field on the associated ThirdPartyAccountLink object is set to false. Querying the ProviderType field on the AuthProvider object sometimes returns a value that differs from the expected provider name value. For example, for Open ID Connect providers, OpenIdConnect is the ProviderType value for the AuthProvider object, but the expected providerName is Open ID Connect.

getAccessToken(authProviderId, providerName)

Returns an access token for the current user using the specified 18-character identifier of an AuthProvider definition in your org and the proper name of the third party, such as Salesforce or Facebook. Note that querying the ProviderType field on the AuthProvider object sometimes returns a value that differs from the expected provider name value. For example, for Open ID Connect providers, OpenIdConnect is the ProviderType value for the AuthProvider object, but the expected providerName is Open ID Connect.

Signature

public static String getAccessToken(String authProviderId, String providerName)

Parameters

authProviderId

Type: String

providerName

Type: String

The proper name of the third party. For all providers except Janrain, the expected values are

- Facebook
- Salesforce
- Open ID Connect
- Microsoft Access Control Service
- LinkedIn
- Twitter
- Google

For Janrain providers, the parameter value is the proper name of the third party used. Yahoo! is an example of a Janrain provider value.

Return Value

Type: String

getAccessTokenMap(authProviderId, providerName)

Returns a map from the third-party identifier to the access token for the currently logged-in Salesforce user. The identifier value depends on the third party. For example, for Salesforce it would be the user ID, while for Facebook it would be the user number. Note that querying the ProviderType field on the AuthProvider object sometimes returns a value that differs from the expected provider name value. For example, for Open ID Connect providers, OpenIdConnect is the ProviderType value for the AuthProvider object, but the expected providerName is Open ID Connect.
Signature

public static Map<String, String> getAccessTokenMap(String authProviderId, String providerName)

Parameters

authProviderId
Type: String

providerName
Type: String

The proper name of the third party. For all providers except Janrain, the expected values are

- Facebook
- Salesforce
- Open ID Connect
- Microsoft Access Control Service
- LinkedIn
- Twitter
- Google

For Janrain providers, the parameter value is the proper name of the third party used. Yahoo! is an example of a Janrain provider value.

Return Value
Type: Map<String, String>

refreshAccessToken(authProviderId, providerName, oldAccessToken)

Returns a map from the third-party identifier containing a refreshed access token for the currently logged-in Salesforce user. Note that querying the ProviderType field on the AuthProvider object sometimes returns a value that differs from the expected provider name value. For example, for Open ID Connect providers, OpenIdConnect is the ProviderType value for the AuthProvider object, but the expected providerName is Open ID Connect.

Signature

public static Map<String, String> refreshAccessToken(String authProviderId, String providerName, String oldAccessToken)

Parameters

authProviderId
Type: String

providerName
Type: String

The proper name of the third party. For all providers except Janrain, the expected values are

- Facebook
For Janrain providers, the parameter value is the proper name of the third party used. Yahoo! is an example of a Janrain provider value.

**oldAccessToken**
Type: String

**Return Value**
Type: Map<String, String>

**Usage**
This method works when using Salesforce or an OpenID Connect provider, but not when using Facebook or Janrain. The returned map contains `AccessToken` and `RefreshError` keys. Evaluate the keys in the response to check if the request was successful. For a successful request, the `RefreshError` value is `null`, and `AccessToken` is a token value. For an unsuccessful request, the `RefreshError` value is an error message, and the `AccessToken` value is `null`.

When successful, this method updates the token stored in the database, which you can get using `Auth.AuthToken.getAccessToken()`.

If you are using an OpenID Connect authentication provider, an `id_token` is not required in the response from the provider. If a `Token Issuer` is specified in the `Auth. Provider` settings and an `id_token` is provided anyway, Salesforce will verify it.

**Example**
```java
String accessToken = Auth.AuthToken.getAccessToken('0SOD000000000DeOAI', 'Open ID connect');
Map<String, String> responseMap = Auth.AuthToken.refreshAccessToken('0SOD000000000DeOAI', 'Open ID connect', accessToken);
```

A successful request includes the access token in the response.
```java
(RefreshError,null)(AccessToken,00DD00000007BhE!AQkAQFzj...)
```

**revokeAccess(authProviderId, providerName, userId, remoteIdentifier)**
Revoke the access token for a specified social sign-on user from a third-party service such as Facebook®. You can use this method only if the `IsNotSsoUsable` field on the associated `ThirdPartyAccountLink` object is set to `false`. Querying the `ProviderType` field on the `AuthProvider` object sometimes returns a value that differs from the expected provider name value. For example, for Open ID Connect providers, `OpenIdConnect` is the `ProviderType` value for the `AuthProvider` object, but the expected `providerName` is `Open ID Connect`.

**Signature**
```java
public static Boolean revokeAccess(String authProviderId, String providerName, String userId, String remoteIdentifier)
```
Parameters

authProviderId
Type: String
The ID of the Auth. Provider in the Salesforce organization.

providerName
Type: String
The proper name of the third party. For all providers except Janrain, the expected values are
- Facebook
- Salesforce
- Open ID Connect
- Microsoft Access Control Service
- LinkedIn
- Twitter
- Google
For Janrain providers, the parameter value is the proper name of the third party used. Yahoo! is an example of a Janrain provider value.

userId
Type: String
The 15-character ID for the user whose access is being revoked.

remoteIdentifier
Type: String
The unique ID for the user in the third-party system (this value is in the associated ThirdPartyAccountLink standard object).

Return Value
Type: Boolean
The return value is true if the revokeAccess() operation is successful; otherwise false.

Example
The following example revokes a Facebook user's access token.

```apex
Auth.AuthToken.revokeAccess('0SOxx00000#####', 'facebook', '005xx00000#####', 'ThirdPartyIdentifier_exist214176560#####');
```

CommunitiesUtil Class
Contains methods for getting information about an Experience Cloud user.

Namespace
Auth
Example

The following example directs a guest (unauthenticated) user to one page, and authenticated users of the Experience Cloud site’s parent organization to another page.

```java
if (Auth.CommunitesUtil.isGuestUser())
  // Redirect to the login page if user is an unauthenticated user
  return new PageReference(LOGIN_URL);

if (Auth.CommunitesUtil.isInternalUser())
  // Redirect to the home page if user is an internal user
  return new PageReference(HOME_URL);
```

CommunitiesUtil Methods

The following are methods for CommunitiesUtil. All methods are static.

**getLogoutUrl()**
Returns the page to display after the current Experience Cloud user logs out.

**getUserDisplayName()**
Returns the current user’s Experience Cloud display name.

**isGuestUser()**
Indicates whether the current user isn’t logged in to the Experience Cloud site. Redirect the user to log in, if necessary.

**isInternalUser()**
Indicates whether the current user is logged in as a member of the parent Salesforce organization, such as an employee.

**getLogoutUrl()**
Returns the page to display after the current Experience Cloud user logs out.

**Signature**

```java
public static String getLogoutUrl()
```

**Return Value**
Type: String

**getUserDisplayName()**
Returns the current user’s Experience Cloud display name.

**Signature**

```java
public static String getUserDisplayName()
```
Return Value
Type: String

`isGuestUser()`
Indicates whether the current user isn't logged in to the Experience Cloud site. Redirect the user to log in, if necessary.

**Signature**
```java
public static Boolean isGuestUser()
```

Return Value
Type: Boolean

`isInternalUser()`
Indicates whether the current user is logged in as a member of the parent Salesforce organization, such as an employee.

**Signature**
```java
public static Boolean isInternalUser()
```

**ConfigurableSelfRegHandler Interface**
Gives you more control over how customers or partners self-register for your Experience Cloud site by creating a class that implements `Auth.ConfigurableSelfRegHandler`. You choose the user information to collect, and how users identify themselves—with their email address, phone number, or another identifier. When verified, you create a customer or partner user and log in the user to your Experience Cloud site.

**Namespace**
`Auth`

**Usage**
You set up site self-registration declaratively on the Login & Registration (L&R) page of the Administration workspace. When combined with a configurable self-registration setup, the handler class can programmatically fill in user fields, including custom fields, and determine how to create a user and log them in.

When you select the Configurable Self-Reg Page registration page, you choose the user fields to collect from the self-registration form, such as last name, first name, username, nickname, mobile, or email. You also determine the verification method that the user identifies themselves with, which can be email, mobile, or neither. Salesforce generates the `Auth.ConfigurableSelfRegHandler` handler, which contains logic on how to create an Experience Cloud site member. Modify the handler to change how users are created, and how collected user information is used.
You can add custom logic to ensure that the email or phone number is unique to the customer or partner who's registering. For example, you can add a custom unique field, and write a copy of the email or phone number to it. You can also change how the user is created. By default, the user is created as a contact associated with the account that you select on the L&R page.

The generated `ConfigurableSelfRegHandler` is located on the Setup Apex Classes page, and begins with `AutocreatedConfigSelfReg`, for example, `AutocreatedConfigSelfReg1532475901849`.

For an example, see `ConfigurableSelfRegHandler Example Implementation`. For more details, see Salesforce Customer Identity in Salesforce Help.

**IN THIS SECTION:**

- **ConfigurableSelfRegHandler Method**
- **ConfigurableSelfRegHandler Example Implementation**

This Apex code implements the `Auth.ConfigurableSelfRegHandler` interface. After the customer or partner fills out the sign-up page and submits it, the handler is invoked to create an Experience Cloud member with the supplied information. If the registration process requires email or phone verification, the verification process finishes before the `Auth.ConfigurableSelfRegHandler.createUser` is invoked. If verification isn't required, `createUser` is invoked when the customer or partner submits the page.

**ConfigurableSelfRegHandler Method**

The following is the method for `ConfigurableSelfRegHandler`.

**IN THIS SECTION:**

- `createUser(accountId, profileId, registrationAttributes, password)`

Create a community member from the information that the visitor provided on your community's self-registration page.

**createUser(accountId, profileId, registrationAttributes, password)**

Create a community member from the information that the visitor provided on your community's self-registration page.

**Signature**

```java
public Id createUser(Id accountId, Id profileId, Map<Schema.SObjectField, String> registrationAttributes, String password)
```

**Parameters**

- **accountId**
  Type: `Id`
  Default account with which the new user is associated. This value comes from the Account field setting on Login and Registration (L&R) page under Registration Page Configuration.

- **profileID**
  Type: `Id`
  Profile to assign the new user. This value comes from the Profile field setting on the L&R page under Registration Page Configuration.

- **registrationAttributes**
  Type: `Map<Schema.sObjectField, String>`
A map of attributes that the registering user entered on the self-registration page. The fields that appear on the self-registration page come from the User Fields selected on the L&R page when the registration type is Configurable Self-Reg Page.

**password**

Type: String

The password entered by the user if “Include Password” is selected on the L&R page. (If a password isn’t entered, the handler must generate one because a password is required to create a user.)

**Return Value**

Type: Id

Returns an identifier for the created User object. Auth.ConfigurableSelfRegHandler inserts a user and then returns the ID of that user.

### ConfigurableSelfRegHandler Example Implementation

This Apex code implements the Auth.ConfigurableSelfRegHandler interface. After the customer or partner fills out the sign-up page and submits it, the handler is invoked to create an Experience Cloud member with the supplied information. If the registration process requires email or phone verification, the verification process finishes before the Auth.ConfigurableSelfRegHandler.createUser is invoked. If verification isn’t required, createUser is invoked when the customer or partner submits the page.

Verification occurs by email if the admin chose Email as the verification method when setting up the Configurable Self-Reg handler on the Login & Registration (L&R) page. When a visitor clicks the sign-up link from the login page, Salesforce prompts for an email address and then sends a one-time password to the specified email address. If the visitor enters the verification code successfully on the verify page, the user is created and logged in. Likewise, if the admin chose Text Message as the verification method on the L&R page, the visitor is prompted to enter a phone number. Salesforce sends a challenge (verification code) via SMS to the user. If successful, the user is created and logged in. Requiring verification before creating a user reduces the number of dummy users cluttering your org.

The Auth.ConfigurableSelfRegHandler class contains logic for generating the user fields required to create a user in case the user doesn’t supply them. The handler generates default values, ensuring that the values are unique by appending a timestamp. You can modify the handler to make sure that the email address and phone number of the customer or partner are also unique.

```apex
global class AutocreatedConfigSelfReg implements Auth.ConfigurableSelfRegHandler {
    private final Long CURRENT_TIME = Datetime.now().getTime();
    private final String[] UPPERCASE_CHARS = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ'.split('');
    private final String[] LOWERCASE_CHARS = 'abcdefghijklmnopqrstuvwxyz'.split('');
    private final String[] NUMBER_CHARS = '1234567890'.split('');
    private final String[] SPECIAL_CHARS = '!#$%-_=+<>'.split('');

    // This method is called once after verification (if any was configured).
    // This method should create a user and insert it.
    // Password can be null.
    // Return null or throw an exception to fail creation.
    global Id createUser(Id accountId, Id profileId, Map<SObjectField, String> registrationAttributes, String password) {
        User u = new User();
        u.ProfileId = profileId;
        for (SObjectField field : registrationAttributes.keySet()) {
            String value = registrationAttributes.get(field);
            u.put(field, value);
        }
    }
}
```
u = handleUnsetRequiredFields(u);
generateContact(u, accountId);
if (String.isBlank(password)) {
    password = generateRandomPassword();
}
Site.validatePassword(u, password, password);
if (u.contactId == null) {
    return Site.createExternalUser(u, accountId, password);
}
u.languagelocalekey = UserInfo.getLocale();
u.localesidkey = UserInfo.getLocale();
u.emailEncodingKey = 'UTF-8';
u.timeZoneSidKey = UserInfo.getTimezone().getID();
insert u;
System.setPassword(u.Id, password);
return u.id;

// Method to autogenerate a password if one isn't passed in.
// By setting a password for a user, we won't send a
// welcome email to set the password.
private String generateRandomPassword() {
    String[] characters = new List<String>(UPPERCASE_CHARS);
    characters.addAll(LOWERCASE_CHARS);
    characters.addAll(NUMBER_CHARS);
    characters.addAll(SPECIAL_CHARS);
    String newPassword = '';
    Boolean needsUpper = true, needsLower = true, needsNumber = true, needsSpecial =
    true;
    while (newPassword.length() < 50) {
        Integer randomInt = generateRandomInt(characters.size());
        String c = characters[randomInt];
        if (needsUpper && c.isAllUpperCase()) {
            needsUpper = false;
        } else if (needsLower && c.isAllLowerCase()) {
            needsLower = false;
        } else if (needsNumber && c.isNumeric()) {
            needsNumber = false;
        } else if (needsSpecial && !c.isAlphanumeric()) {
            needsSpecial = false;
        } else {
            newPassword += c;
        }
    }
    newPassword = addMissingPasswordRequirements(newPassword, needsLower, needsUpper,
    needsNumber, needsSpecial);
    return newPassword;
}

private String addMissingPasswordRequirements(String password, Boolean addLowerCase,
    Boolean addUpperCase, Boolean addNumber, Boolean addSpecial) {
    if (addLowerCase) {
        password += LOWERCASE_CHARS[generateRandomInt(LOWERCASE_CHARS.size())];
    }
    if (addUpperCase) {
        password += UPPERCASE_CHARS[generateRandomInt(UPPERCASE_CHARS.size())];
    }
    if (addNumber) {
        password += NUMBER_CHARS[generateRandomInt(NUMBER_CHARS.size())];
    }
    if (addSpecial) {
        password += SPECIAL_CHARS[generateRandomInt(SPECIAL_CHARS.size())];
    }
    return password;  
}
password += UPPERCASE_CHARS[generateRandomInt(UPPERCASE_CHARS.size())];
if (addNumber) {
    password += NUMBER_CHARS[generateRandomInt(NUMBER_CHARS.size())];
} if (addSpecial) {
    password += SPECIAL_CHARS[generateRandomInt(SPECIAL_CHARS.size())];
} return password;

private Integer generateRandomInt(Integer max) {
    return Math.mod(Math.abs(Crypto.getRandomInteger()), max);
}

private User handleUnsetRequiredFields(User u) {
    if (String.isBlank(u.LastName)) {
        u.LastName = generateLastName();
    }
    if (String.isBlank(u.Username)) {
        u.Username = generateUsername();
    }
    if (String.isBlank(u.Email)) {
        u.Email = generateEmail();
    }
    if (String.isBlank(u.Alias)) {
        u.Alias = generateAlias();
    }
    if (String.isBlank(u.CommunityNickname)) {
        u.CommunityNickname = generateCommunityNickname();
    }
    return u;
}

private void generateContact(User u, Id accountId) {
    // Add logic here if you want to build your own // contact for the use.
}

private String generateAlias() {
    String timeString = String.valueOf(CURRENT_TIME);
    return timeString.substring(timeString.length() - 8);
}

private String generateLastName() {
    return 'ExternalUser' + CURRENT_TIME;
}

private String generateUsername() {
    return 'externaluser' + CURRENT_TIME + '@company.com';
}

// Default implementation to try to provide uniqueness.
private String generateEmail() {
    return 'externaluser' + CURRENT_TIME + '@company.com';
}

// Default implementation to try to provide uniqueness.
private String generateCommunityNickname() {
    return 'ExternalUser' + CURRENT_TIME;
}

ConfirmUserRegistrationHandler Interface

Manages single sign-on (SSO) user mappings between Salesforce and a third-party identity provider. Use this interface to confirm user mappings before updating them.

Namespace

Auth

Usage

When you set up SSO with a third-party identity provider, you create a class that implements a registration handler using the Auth.RegistrationHandler interface. This class manages the process of creating and updating users. For advanced use cases that require you to confirm user information during the update process, implement the Auth.ConfirmUserRegistrationHandler interface in your class. This interface must be implemented in addition to Auth.RegistrationHandler.

You can use the Auth.ConfirmUserRegistrationHandler interface to ensure that users are mapped correctly between Salesforce and the third party. When a user who has previously logged in with an authentication provider logs in again, you can confirm that the incoming user data is consistent with the user’s third-party identifier. If not, you can identify which user is supposed to be logged in.

You can also use the Auth.ConfirmUserRegistrationHandler interface to switch context for users with multiple records. For example, a user has two records—an admin user and a standard user. When the user logs in, the third-party identity provider confirms the account used to log in and sends the response to Salesforce via the UserInfo endpoint. You can then use this information to determine whether to log in the user as an admin or standard user.

IN THIS SECTION:

ConfirmUserRegistrationHandler Methods

ConfirmUserRegistrationHandler Example Implementation

ConfirmUserRegistrationHandler Methods

The following are methods for ConfirmUserRegistrationHandler.
IN THIS SECTION:

confirmUser(userId, tpalId, portalId, userdata)

Returns the ID of the user to be logged in based on their mapping to a third-party identifier. This method is called before calling the updateUser() method. It’s called only if the incoming user has previously logged in and has a third-party account link to a Salesforce user.

**confirmUser(userId, tpalId, portalId, userdata)**

Returns the ID of the user to be logged in based on their mapping to a third-party identifier. This method is called before calling the updateUser() method. It’s called only if the incoming user has previously logged in and has a third-party account link to a Salesforce user.

**Signature**

```java
public Id confirmUser(Id userId, Id tpalId, Id portalId, Auth.UserData userdata)
```

**Parameters**

- **userId**
  
  Type: Id
  
  The ID of the user who is mapped to the third-party identifier via a third-party account link.

- **tpalId**
  
  Type: Id
  
  The third-party account link corresponding to the third-party identifier.

- **portalId**
  
  Type: Id
  
  The portal ID the user is logging in to. If there’s no portal configured, this value can be null.

- **userData**
  
  Type: Auth.UserData
  
  Contains user information from the third-party identity provider.

**Return Value**

Type: Id

The Id of the user to be logged in. If null, login fails.

**ConfirmUserRegistrationHandler Example Implementation**

This example implements the Auth.ConfirmUserRegistrationHandler interface during the user update process to confirm that the correct user is logging in based on their email address and last name.

```java
global class StandardUserRegistrationHandler implements Auth.RegistrationHandler,
Auth.ConfirmUserRegistrationHandler {
    global User createUser(Id portalId, Auth.UserData data){
        User u = new User();
        Profile p = [SELECT Id FROM profile WHERE name='Standard User'];
        u.username = data.username + '@salesforce.com';
    }
}
```
global void updateUser(Id userId, Id portalId, Auth.UserData data) {
    User u = new User(id=userId);
    u.username = data.username + '@salesforce.com';
    u.email = data.email;
    u.lastName = data.lastName;
    u.firstName = data.firstName;
    String alias = data.username;
    if(alias.length() > 8) {
        alias = alias.substring(0, 8);
    }
    u.alias = alias;
    u.languagelocalekey = data.attributeMap.get('language');
    u.localesidkey = data.locale;
    u.emailEncodingKey = 'UTF-8';
    u.timeZoneSidKey = 'America/Los_Angeles';
    u.profileId = p.Id;
    return u;
}

global Id confirmUser(Id userId, Id tpalId, Id portalId, Auth.UserData data) {
    if (data.email.contains(data.lastName)) { // looks genuine
        return userId;
    } else { // find the right user
        User confirmedUser = [SELECT id FROM user WHERE email=:data.email];
        return confirmedUser.Id;
    }
}

The following example tests the implementation:

@isTest
class StandardUserRegistrationHandlerTest {
    static testMethod void testConfirmUser() {
        StandardUserRegistrationHandler handler = new StandardUserRegistrationHandler();
        Auth.UserData sampleData = new Auth.UserData('idA', 'firstName', 'A', 'firstName A', 'userA@example.org', null, 'usernameA', 'en_US', 'facebook', null, new Map<String, String>{'language' => 'en_US'});
        User u = handler.createUser(null, sampleData);
        insert(u);
        String uid = u.id;
}
sampleData = new Auth.UserData('idB', 'firstName', 'B',
    'firstName B', 'userA@example.org', null, 'usernameB', 'en_US', 'facebook',
    null, new Map<String, String>{}); // note that user B is using userA's email
Id confirmedUserId = handler.confirmUser(uid, '060xx0000004Eh6', null, sampleData);
System.assertEquals(uid, confirmedUserId); // we should see userA's id
}

ConnectedAppPlugin Class

Contains methods for extending the behavior of a connected app, for example, customizing how a connected app is invoked depending on the protocol used. This class gives you more control over the interaction between Salesforce and your connected app.

Namespace

Auth

Usage

When you create a connected app, you specify general information about the app and settings for OAuth, web apps, mobile apps, and canvas apps. To customize how the app is invoked, create a connected app handler with this ConnectedAppPlugin Apex class. For example, use this class to support new authentication protocols or respond to user attributes in a way that benefits a business process.

When you create a connected app handler, you also configure the ConnectedAppPlugin class to run as an execution user. The execution user authorizes access for the connected app. For example, when you use the authorize method, the execution user authorizes the connected app to access data.

If you don’t specify an execution user, the plug-in runs as an Automated Process User, which is a system user that executes tasks behind the scenes. Most ConnectedAppPlugin methods require that you specify an execution user, with the exception of the customAttributes method. For more information, see Create a Custom Connected App Handler.

Example

This example authorizes the connected app user to use the connected app if the context is SAML and the user has reached the quota tracked in a custom field. It returns the user’s permission set assignments. The example uses Auth.InvocationContext to modify a SAML assertion before it’s sent to the service provider.

global class ConnectedAppPluginExample extends Auth.ConnectedAppPlugin
{
    // Authorize the app if the user has achieved quota tracked in a custom field
    global override Boolean authorize(Id userId, Id connectedAppId, Boolean isAdminApproved, Auth.InvocationContext context)
    {
        // Create a custom boolean field HasAchievedQuota__c on the user record
        // and then uncomment the block below
        // User u = [select id, HasAchievedQuota__c from User where id =: userId].get(0);
        // return u.HasAchievedQuota__c;

        return isAdminApproved;
    }
}
// Call a flow during refresh
global override void refresh(Id userId, Id connectedAppId, Auth.InvocationContext context)
{
  try
  {
    Map<String, Object> inputVariables = new Map<String, Object>();
    inputVariables.put('userId', userId);
    inputVariables.put('connectedAppId', connectedAppId);

    // Create a custom trigger ready flow and uncomment the block below
    // Flow.Interview.MyCustomFlow interview = new Flow.Interview.MyCustomFlow(inputVariables);
    // interview.start();
  }
  catch (Exception e) {
    System.debug('FLOW Exception:' + e);
  }
}

// Return a user's permission set assignments
global override Map<String,String> customAttributes(Id userId, Id connectedAppId,
Map<String,String> formulaDefinedAttributes, Auth.InvocationContext context)
{
  List<PermissionSetAssignment> psas = [SELECT id, PermissionSet.Name FROM PermissionSetAssignment
WHERE PermissionSet.IsOwnedByProfile = false AND (AssigneeId = :userId)];
  String permsets = '][';
  for (PermissionSetAssignment psa :psas)
  {
    permsets += psa.PermissionSet.Name + ';';
  }
  permsets += ']]';
  formulaDefinedAttributes.put('PermissionSets', permsets);
  return formulaDefinedAttributes;
}

IN THIS SECTION:

ConnectedAppPlugin Methods

ConnectedAppPlugin Methods
The following are methods for ConnectedAppPlugin.

IN THIS SECTION:

authorize(userId, connectedAppId, isAdminApproved)
Deprecated and available only in API versions 35.0 and 36.0. As of version 37.0, use authorize (userId, connectedAppId, isAdminApproved, context) instead.
authorize(userId, connectedAppId, isAdminApproved, context)
Authorizes the specified user to access the connected app. If the connected app is set for users to self-authorize, this method isn’t invoked.

customAttributes(userId, connectedAppId, formulaDefinedAttributes)
Deprecated and available only in API versions 35.0 and 36.0. As of version 37.0, use customAttributes(userId, connectedAppId, formulaDefinedAttributes, context) instead.

customAttributes(userId, connectedAppId, formulaDefinedAttributes, context)
Sets new attributes for the specified user. When the connected app gets the user’s attributes from the UserInfo endpoint or through a SAML assertion, use this method to update the attribute values.

modifySAMLResponse(authSession, connectedAppId, samlResponse)
Modifies the XML generated by the Salesforce SAML Identity Provider (IDP) before it’s sent to the service provider.

refresh(userId, connectedAppId)
Deprecated and available only in API versions 35.0 and 36.0. As of version 37.0, use refresh(userId, connectedAppId, context) instead.

refresh(userId, connectedAppId, context)
Salesforce calls this method during a refresh token exchange.

authorize(userId, connectedAppId, isAdminApproved)
Deprecated and available only in API versions 35.0 and 36.0. As of version 37.0, use authorize(userId, connectedAppId, isAdminApproved, context) instead.

Signature

public Boolean authorize(Id userId, Id connectedAppId, Boolean isAdminApproved)

Parameters

userId
Type: Id
The 15-character ID of the user attempting to use the connected app.

connectedAppId
Type: String
The 15-character ID of the connected app.

isAdminApproved
Type: Boolean
The approval state of the specified user when the connected app requires approval.

Return Value

Type: Boolean
If the connected app requires admin approval, a returned value of true indicates that the current user is approved.
authorize(userId, connectedAppId, isAdminApproved, context)

Authorizes the specified user to access the connected app. If the connected app is set for users to self-authorize, this method isn’t invoked.

Signature

```
public Boolean authorize(Id userId, Id connectedAppId, Boolean isAdminApproved, 
Auth.InvocationContext context)
```

Parameters

- **userId**
  - Type: Id
  - The 15-character ID of the user attempting to use the connected app.

- **connectedAppId**
  - Type: Id
  - The 15-character ID of the connected app.

- **isAdminApproved**
  - Type: Boolean
  - The approval state of the specified user when the connected app requires approval.

- **context**
  - Type: InvocationContext
  - The context in which the connected app is invoked.

Return Value

- Type: Boolean
  - If the connected app requires admin approval, a returned value of `true` indicates that the user is approved.

Usage

`ConnectedAppPlugin` runs on behalf of the current user. But the user must have permission to use the connected app for the plug-in to work. Use this method to authorize the user.

customAttributes(userId, connectedAppId, formulaDefinedAttributes)

Deprecated and available only in API versions 35.0 and 36.0. As of version 37.0, use `customAttributes(userId, connectedAppId, formulaDefinedAttributes, context)` instead.

Signature

```
public Map<String,String> customAttributes(Id userId, Id connectedAppId, 
Map<String,String> formulaDefinedAttributes,)
```

Parameters

- **userId**
  - Type: Id
The 15-character ID of the user attempting to use the connected app.

`connectedAppId`
Type: Id
The 15-character ID of the connected app.

`formulaDefinedAttributes`
Type: Map<String,String>
A map of the new set of attributes from the UserInfo endpoint (OAuth) or from a SAML assertion. For more information, see The UserInfo Endpoint in the online help.

**Return Value**
Type: Map<String,String>
A map of the updated set of attributes.

`customAttributes(userId, connectedAppId, formulaDefinedAttributes, context)`
Sets new attributes for the specified user. When the connected app gets the user’s attributes from the UserInfo endpoint or through a SAML assertion, use this method to update the attribute values.

**Signature**

```java
public Map<String,String> customAttributes(Id userId, Id connectedAppId, Map<String,String> formulaDefinedAttributes, Auth.InvocationContext context)
```

**Parameters**

`userId`
Type: Id
The 15-character ID of the user attempting to use the connected app.

`connectedAppId`
Type: Id
The 15-character ID for the connected app.

`formulaDefinedAttributes`
Type: Map<String,String>
A map of the current set of attributes from the UserInfo endpoint (OAuth) or from a SAML assertion. For more information, see The UserInfo Endpoint in the online help.

`context`
Type: InvocationContext
The context in which the connected app is invoked.

**Return Value**
Type: Map<String,String>
A map of the updated set of attributes.
modifySAMLResponse(authSession, connectedAppId, samlResponse)

Modifies the XML generated by the Salesforce SAML Identity Provider (IDP) before it's sent to the service provider.

Signature

```java
public dom.XmlNode modifySAMLResponse(Map<String,String> authSession, Id connectedAppId, dom.XmlNode samlResponse)
```

Parameters

- `authSession` Type: `Map<String,String>`
  The attributes for the authorized user's session. The map includes the 15-character ID of the authorized user who's accessing the connected app.

- `connectedAppId` Type: `Id`
  The 15-character ID of the connected app.

- `samlResponse` Type: `Dom.XmlNode`
  Contains the SAML XML response generated by the IDP.

Return Value

Type: `Dom.XmlNode`

Returns an instance of `Dom.XmlNode` containing the modified SAML XML response.

Usage

Use this method to modify the XML SAML response to perform an action based on the context of the SAML request before it's verified, signed, and sent to the target service provider. This method enables developers to extend the connected app plug-in to meet their specific needs.

The developer assumes full responsibility for changes made within the connected app plug-in. The plug-in must include validation and error handling. If the plug-in throws an exception, catch it, log it, and stop the process. Don't send anything to the target service provider.

refresh(userId, connectedAppId)

Deprecated and available only in API versions 35.0 and 36.0. As of version 37.0, use `refresh(userId, connectedAppId, context)` instead.

Signature

```java
public void refresh(Id userId, Id connectedAppId)
```

Parameters

- `userId` Type: `Id`
  

108
The 15-character ID of the user requesting the refresh token.

\texttt{connectedAppId}
\begin{itemize}
  \item \textbf{Type: \textit{Id}}
  \item The 15-character ID of the connected app.
\end{itemize}

\textbf{Return Value}
\begin{itemize}
  \item \textbf{Type: void}
\end{itemize}

\texttt{refresh(userId, connectedAppId, context)}
Salesforce calls this method during a refresh token exchange.

\textbf{Signature}
\begin{itemize}
  \item \texttt{public void refresh(Id userId, Id connectedAppId, Auth.InvocationContext context)}
\end{itemize}

\textbf{Parameters}
\begin{itemize}
  \item \texttt{userId}
    \begin{itemize}
      \item \textbf{Type: \textit{Id}}
      \item The 15-character ID of the user requesting the refresh token.
    \end{itemize}
  \item \texttt{connectedAppId}
    \begin{itemize}
      \item \textbf{Type: \textit{Id}}
      \item The 15-character ID of the connected app.
    \end{itemize}
  \item \texttt{context}
    \begin{itemize}
      \item \textbf{Type: InvocationContext}
      \item The context in which the connected app is invoked.
    \end{itemize}
\end{itemize}

\textbf{Return Value}
\begin{itemize}
  \item \textbf{Type: void}
\end{itemize}

\textbf{HeadlessSelfRegistrationHandler Interface}
Creates customer and partner users during the Headless Registration Flow.

\textbf{Namespace}
\begin{itemize}
  \item \texttt{Auth}
\end{itemize}

\textbf{Usage}
The Headless Registration Flow allows you to control user registration experience in a third-party app while using Salesforce to authenticate users and manage their data access. When you set up this flow, add users in the class that is implementing the \texttt{Auth.HeadlessSelfRegistrationHandler} interface. This class runs after the user verifies their identity. For a detailed
explanation of headless registration, see Headless Registration Flow for Private Clients or Headless Registration Flow for Public Clients, depending on your app type.

IN THIS SECTION:

HeadlessSelfRegistrationHandler Methods
The following are methods for HeadlessSelfRegistrationHandler.

HeadlessSelfRegistrationHandler Example Implementation
This example class implements the Auth.HeadlessSelfRegistrationHandler interface. The class creates a user based on information that your app sends to Headless Registration API. It associates the user with the default profile for new users, which you configure in your Experience Cloud settings. It also creates a contact for the user and assigns it to an account.

HeadlessSelfRegistrationHandler Methods
The following are methods for HeadlessSelfRegistrationHandler.

IN THIS SECTION:

createUser(profileId, data, customUserDataMap, experienceId, password)
Returns a User object using information submitted by your off-platform app to Headless Registration API. The User object can be a new user that hasn’t been inserted in your org’s database, or it can represent an existing user record. If it’s a new User object, Salesforce inserts the user record for you.

createUser (profileId, data, customUserDataMap, experienceId, password)
Returns a User object using information submitted by your off-platform app to Headless Registration API. The User object can be a new user that hasn’t been inserted in your org’s database, or it can represent an existing user record. If it’s a new User object, Salesforce inserts the user record for you.

Signature
public User createUser(Id profileId, Auth.UserData data, String customUserDataMap, String experienceId, String password)

Parameters

profileId
Type: Id
The ID of the profile that is assigned to new users.

data
Type: Auth.UserData
A class that stores information about the user, such as their name and locale.

customUserDataMap
Type: String
A string representation of a JSON object containing custom user information passed in during registration. We recommend that you deserialize this string into the equivalent Apex class structure. Determine what custom information to collect when you build your app’s registration experience.
**experienceId**
Type: String
The Experience Cloud site ID.

**password**
Type: String
The user password.

**Return Value**
Type: User

### HeadlessSelfRegistrationHandler Example Implementation

This example class implements the `Auth.HeadlessSelfRegistrationHandler` interface. The class creates a user based on information that your app sends to Headless Registration API. It associates the user with the default profile for new users, which you configure in your Experience Cloud settings. It also creates a contact for the user and assigns it to an account.

```apex
global class ExampleHeadlessReg implements Auth.HeadlessSelfRegistrationHandler {

    global User createUser(Id profileId, Auth.UserData data, String customUserDataMap, String experienceId, String password){
        User u = new User();
        u.ProfileId = profileId;
        u.email = data.email;
        Account a;
        List<Account> accounts = 
            [SELECT Id FROM account WHERE name=:social_account];
        if(accounts.isEmpty())
        {
            a = new Account(name = social_account);
            insert(a);
        }else
        a = accounts[0];
        generateContact(u, a.Id);
        u.languagelocalekey = UserInfo.getLocale();
        u.localesidkey = UserInfo.getLocale();
        u.timeZoneSidKey = UserInfo.getTimezone().getID();
        return u;
    }

    private User handleUnsetRequiredFields(User u) {
        return u;
    }

    // Method to construct a contact for a user
    private void generateContact(User u, Id accountId) {
        // Add logic here to build your own contact for the user
        Contact c = new Contact();
        c.accountId = accountId;
    }

```
InvocationContext Enum

The context in which the connected app is invoked, such as the protocol flow used and the token type issued, if any. Developers can use the context information to write code that is unique to the type of invocation.

Enum Values

The following are the values of the Auth.InvocationContext enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASSET_TOKEN</td>
<td>Reserved for future use.</td>
</tr>
<tr>
<td>OAUTH1</td>
<td>Context used when authentication is through an OAuth 1.0A flow.</td>
</tr>
<tr>
<td>OAUTH2_JWT_BEARER_TOKEN</td>
<td>Context used when authentication is through a JSON-based Web Token (JWT) bearer token flow.</td>
</tr>
<tr>
<td>OAUTH2_SAML_ASSERTION</td>
<td>Context used when authentication is through an OAuth 2.0 SAML assertion flow.</td>
</tr>
<tr>
<td>OAUTH2_SAML_BEARER_ASSERTION</td>
<td>Context used when authentication is through an OAuth 2.0 SAML bearer assertion flow.</td>
</tr>
<tr>
<td>OAUTH2_USERNAME_PASSWORD</td>
<td>Context used when authentication is through an OAuth 2.0 username-password flow.</td>
</tr>
<tr>
<td>OAUTH2_USER_AGENT_ID_TOKEN</td>
<td>Context used when issuing an ID token through an OAuth 2.0 user-agent flow.</td>
</tr>
<tr>
<td>OAUTH2_USER_AGENT_TOKEN</td>
<td>Context used when authentication is through an OAuth 2.0 user agent flow.</td>
</tr>
<tr>
<td>OAUTH2_WEB_SERVER</td>
<td>Context used when authentication is through a web server authentication flow.</td>
</tr>
<tr>
<td>OPENIDCONNECT</td>
<td>Context used when authentication is through an OpenID Connect authentication flow.</td>
</tr>
<tr>
<td>REFRESH_TOKEN</td>
<td>Context used when renewing tokens issued by a web server or user-agent flow.</td>
</tr>
<tr>
<td>SAML_ASSERTION</td>
<td>Context used when authentication is through a SAML assertion flow.</td>
</tr>
<tr>
<td>UNKNOWN</td>
<td>Context is unknown.</td>
</tr>
</tbody>
</table>
USERID_ENDPOINT | Context used when issuing an access token through a UserInfo endpoint.

SEE ALSO:
Salesforce Help: Authenticating Apps with OAuth

### JWS Class

Contains methods that apply a digital signature to a JSON Web Token (JWT), using a JSON Web Signature (JWS) data structure. This class creates the signed JWT bearer token, which can be used to request an OAuth access token in the OAuth 2.0 JWT bearer token flow.

### Namespace

Auth

### Usage

Use the methods in this class to sign the JWT bearer token with the X509 certificate.

### IN THIS SECTION:

- JWS Constructors
- JWS Methods

### JWS Constructors

The following are constructors for JWS.

### IN THIS SECTION:

- JWS(jwt, certDevName)
  Creates an instance of the JWS class using the specified Auth.JWT payload and the certificate used for signing the JWT bearer token.
- JWS(payload, certDevName)
  Creates an instance of the JWS class using the specified payload and certificate used for signing the JWT bearer token.

### JWS(jwt, certDevName)

Creates an instance of the JWS class using the specified Auth.JWT payload and the certificate used for signing the JWT bearer token.

### Signature

```java
public JWS(Auth.JWT jwt, String certDevName)
```
Parameters

jwt
  Type: Auth.JWT
  The Base64-encoded JSON Claims Set in the JWT bearer token generated by Auth.JWT.

certDevName
  Type: String
  The Unique Name for a certificate stored in the Salesforce org’s Certificate and Key Management page to use for signing the JWT bearer token.

Usage

Calls the toJSONString() method in Auth.JWT and sets the resulting string as the payload of the JWT bearer token. Alternatively, you can specify the payload directly using JWS(payload, certDevName).

JWS(payload, certDevName)

Creates an instance of the JWS class using the specified payload and certificate used for signing the JWT bearer token.

Signature

public JWS(String payload, String certDevName)

Parameters

payload
  Type: String
  The Base64-encoded JSON Claims Set in the JWT bearer token.

certDevName
  Type: String
  The Unique Name for a certificate stored in the Salesforce org’s Certificate and Key Management page to use for signing the JWT bearer token.

Usage

Sets the payload string as the payload of the JWT bearer token. Alternatively, if you generate the payload using Auth.JWT, you can use JWS(jwt, certDevName) instead.

JWS Methods

The following are methods for JWS. All are instance methods.

IN THIS SECTION:

  clone()
  Makes a duplicate copy of the JWS object.
getCompactSerialization()
Returns the compact serialization representation of the JWS as a concatenated string, with the encoded JWS header, encoded JWS payload, and encoded JWS signature strings separated by period (\.') characters.

clone()
Makes a duplicate copy of the JWS object.

Signature
public Object clone()

Return Value
Type: JWS

getCompactSerialization()
Returns the compact serialization representation of the JWS as a concatenated string, with the encoded JWS header, encoded JWS payload, and encoded JWS signature strings separated by period (\.') characters.

Signature
public String getCompactSerialization()

Return Value
Type: String

JWT Class
Generates the JSON Claims Set in a JSON Web Token (JWT). The resulting Base64-encoded payload can be passed as an argument to create an instance of the Auth.JWS class.

Namespace
Auth

Usage
Use the methods in this class to generate the payload in a JWT bearer token.

IN THIS SECTION:
JWT Methods

JWT Methods
The following are methods for JWT. All are instance methods.
IN THIS SECTION:

clone()
Makes a duplicate copy of the JWT object.

getAdditionalClaims()
Returns a map of additional claims in the JWT, where the key string contains the name of the claim, and the value contains the value of the claim.

getAud()
Returns the audience claim that identifies the intended recipients of the JWT.

getIss()
Returns the issuer claim that identifies the issuer of the JWT.

getNbfClockSkew()
Returns the not before claim that identifies the time before which the JWT must not be accepted for processing, while allowing some leeway for clock skew.

getSub()
Returns the subject claim that identifies the current user of the JWT.

getValidityLength()
Returns the length of time (in seconds) that the JWT is valid, which affects the expiration claim.

setAdditionalClaims(additionalClaims)
Sets the additional claims in the JWT. Returned by the getAdditionalClaims() method.

setAud(aud)
Sets the audience claim in the JWT. Returned by the getAud() method.

setIss(iss)
Sets the issuer claim in the JWT. Returned by the getIss() method.

setNbfClockSkew(nbfClockSkew)
Sets the not before claim in the JWT. Returned by the getNbfClockSkew() method.

setSub(sub)
Sets the subject claim in the JWT. Returned by the getSub() method.

setValidityLength(validityLength)
Sets the length of time (in seconds) that the JWT is valid, which affects the expiration claim. Returned by the getValidityLength() method.

toJsonString()
Generates the JSON object representation of the Claims Set as an encoded JWT payload.

clone()
Makes a duplicate copy of the JWT object.

Signature

public Object clone()
Return Value
type: JWT

`getAdditionalClaims()`
Returns a map of additional claims in the JWT, where the key string contains the name of the claim, and the value contains the value of the claim.

**Signature**
```
public Map<String, ANY> getAdditionalClaims()
```

Return Value
type: Map<String, ANY>

`getAud()`
Returns the audience claim that identifies the intended recipients of the JWT.

**Signature**
```
public String getAud()
```

Return Value
type: String

`getIss()`
Returns the issuer claim that identifies the issuer of the JWT.

**Signature**
```
public String getIss()
```

Return Value
type: String

`getNbfClockSkew()`
Returns the not before claim that identifies the time before which the JWT must not be accepted for processing, while allowing some leeway for clock skew.

**Signature**
```
public Integer getNbfClockSkew()
```
Return Value
Type: Integer

`getSub()`
Returns the subject claim that identifies the current user of the JWT.

Signature
`public String getSub()`

Return Value
Type: String

`getValidityLength()`
Returns the length of time (in seconds) that the JWT is valid, which affects the expiration claim.

Signature
`public Integer getValidityLength()`

Return Value
Type: Integer

`setAdditionalClaims(additionalClaims)`
Sets the additional claims in the JWT. Returned by the `getAdditionalClaims()` method.

Signature
`public void setAdditionalClaims(Map<String,ANY> additionalClaims)`

Parameters
`additionalClaims`
Type: `Map<String,ANY>`

Return Value
Type: void

Usage
Additional claims must not include any standard claims.
**setAud(aud)**
Sets the audience claim in the JWT. Returned by the `getAud()` method.

**Signature**
```java
public void setAud(String aud)
```

**Parameters**

*aud*
Type: `String`

**Return Value**
Type: `void`

**setIss(iss)**
Sets the issuer claim in the JWT. Returned by the `getIss()` method.

**Signature**
```java
public void setIss(String iss)
```

**Parameters**

*iss*
Type: `String`

**Return Value**
Type: `void`

**setNbfClockSkew(nbfClockSkew)**
Sets the not before claim in the JWT. Returned by the `getNbfClockSkew()` method.

**Signature**
```java
public void setNbfClockSkew(Integer nbfClockSkew)
```

**Parameters**

*nbfClockSkew*
Type: `Integer`

**Return Value**
Type: `void`
**setSub (sub)**
Sets the subject claim in the JWT. Returned by the `getSub()` method.

**Signature**
```
public void setSub(String sub)
```

**Parameters**

*sub*
Type: `String`

**Return Value**
Type: `void`

**setValidityLength (validityLength)**
Sets the length of time (in seconds) that the JWT is valid, which affects the expiration claim. Returned by the `getValidityLength()` method.

**Signature**
```
public void setValidityLength(Integer validityLength)
```

**Parameters**

*validityLength*
Type: `Integer`

**Return Value**
Type: `void`

**toJSONString()**
Generates the JSON object representation of the Claims Set as an encoded JWT payload.

**Signature**
```
public String toJSONString()
```

**Return Value**
Type: `String`

**JWTBearerTokenExchange Class**
Contains methods that POST the signed JWT bearer token to a token endpoint to request an access token, in the OAuth 2.0 JWT bearer token flow.
Namespace

Auth

Usage

Use the methods in this class to post a signed JWT bearer token to the OAuth token endpoint, in exchange for an access token.

Example

In the following example application, the Apex controller:

1. Creates the JSON Claims Set.
2. Specifies the scope of the request with additional claims.
3. Creates the signed JWT.
4. Specifies the token endpoint and POSTs to it.
5. Gets the access token from the HTTP response.

```java
public class MyController{

    public MyController() {
        Auth.JWT jwt = new Auth.JWT();
        jwt.setSub('user@salesforce.com');
        jwt.setAud('https://login.salesforce.com');
        jwt.setIss('3MVG99OxTyEMCQ3gNp2PjkqeZKxnmA1GLxV4oHh9AKL_rSK.BoSVFGZHQukXnVjzRgSuQqGn75NL7ykQcyy7');

        //Additional claims to set scope
        Map<String, Object> claims = new Map<String, Object>();
        claims.put('scope', 'scope name');
        jwt.setAdditionalClaims(claims);

        //Create the object that signs the JWT bearer token
        Auth.JWS jws = new Auth.JWS(jwt, 'CertFromCertKeyManagement');

        //Get the resulting JWS in case debugging is required
        String token = jws.getCompactSerialization();

        //Set the token endpoint that the JWT bearer token is posted to
        String tokenEndpoint = 'https://login.salesforce.com/services/oauth2/token';

        //POST the JWT bearer token
        Auth.JWTBearerTokenExchange bearer = new Auth.JWTBearerTokenExchange(tokenEndpoint, jws);

        //Get the access token
        String accessToken = bearer.getAccessToken();
    }
}
```
JWTBearerTokenExchange Constructors

The following are constructors for JWTBearerTokenExchange.

IN THIS SECTION:

JWTBearerTokenExchange Constructors

JWTBearerTokenExchange(tokenEndpoint, jws)

Creates an instance of the JWTBearerTokenExchange class using the specified token endpoint and the signed JWT bearer token.

JWTBearerTokenExchange()

Creates an instance of the Auth.JWTBearerTokenExchange class.

Signature

public JWTBearerTokenExchange(String tokenEndpoint, Auth.JWS jws)

Parameters

tokenEndpoint
Type: String
The token endpoint that the signed JWT bearer token is POSTed to.

jws
Type: Auth.JWS
The signed JWT bearer token.

JWTBearerTokenExchange methods

The following are methods for JWTBearerTokenExchange. All are instance methods.

Signature

public JWTBearerTokenExchange()
IN THIS SECTION:

- **clone()**
  Makes a duplicate copy of the JWTBearerTokenExchange object.

- **getAccessToken()**
  Returns the access_token in the token response to the JWT bearer token request.

- **getGrantType()**
  Returns the grant type specified in the JWT bearer token request. The grant type value defaults to `urn:ietf:params:oauth:grant-type:jwt-bearer`.

- **getHttpResponse()**
  Returns the full `System.HttpResponse` token response to the JWT bearer token request.

- **getJWS()**
  Returns the JWS specified in the JWT bearer token request.

- **getTokenEndpoint()**
  Returns the token endpoint that the JWT bearer token request is POSTed to.

- **setGrantType(grantType)**
  Sets the grant type in the JWT bearer token request. Returned by the `getGrantType()` method.

- **setJWS(jws)**
  Sets the JWS in the JWT bearer token request. Returned by the `getJWS()` method.

- **setTokenEndpoint(tokenEndpoint)**
  Sets the token endpoint that the JWT bearer token request is POSTed to. Returned by the `getTokenEndpoint()` method.

---

**clone()**

Makes a duplicate copy of the JWTBearerTokenExchange object.

**Signature**

```java
public Object clone()
```

**Return Value**

Type: **JWTBearerTokenExchange**

**getAccessToken()**

Returns the access_token in the token response to the JWT bearer token request.

**Signature**

```java
public String getAccessToken()
```

**Return Value**

Type: **String**
Usage
This method extracts the access_token from the token response. If the token response issues the access token in a different parameter, the request fails.
If you want the full HTTP token response returned, use getHttpResponse instead.

getGrantType()
Returns the grant type specified in the JWT bearer token request. The grant type value defaults to urn:ietf:params:oauth:grant-type:jwt-bearer.

Signature
public String getGrantType()

Return Value
Type: String

getHttpResponse()
Returns the full System.HttpResponse token response to the JWT bearer token request.

Signature
public System.HttpResponse getHttpResponse()

Return Value
Type: System.HttpResponse

Usage
You can get the access token from the full System.HttpResponse. If you want only the access_token from the token response, you can use getAccessToken instead.

getJWS()
Returns the JWS specified in the JWT bearer token request.

Signature
public Auth.JWS getJWS()

Return Value
Type: Auth.JWS

getTokenEndpoint()
Returns the token endpoint that the JWT bearer token request is POSTed to.
Signature

public String getTokenEndpoint()

Return Value
Type: String

`setGrantType(grantType)`
Sets the grant type in the JWT bearer token request. Returned by the `getGrantType()` method.

Signature

public void setGrantType(String grantType)

Parameters

`grantType`  
Type: String

Return Value
Type: void

`setJWS(jws)`
Sets the JWS in the JWT bearer token request. Returned by the `getJWS()` method.

Signature

public void setJWS(Auth.JWS jws)

Parameters

`jws`  
Type: Auth.JWS

Return Value
Type: void

`setTokenEndpoint(tokenEndpoint)`
Sets the token endpoint that the JWT bearer token request is POSTed to. Returned by the `getTokenEndpoint()` method.

Signature

public void setTokenEndpoint(String tokenEndpoint)
Parameters

tokenEndpoint
  Type: String

Return Value
  Type: void

LightningLoginEligibility Enum

Contains a Lightning Login eligibility value used by the Auth.SessionManagement.getLightningLoginEligibility method.

Usage

If you use the Discovery page type, users can verify themselves with Lightning Login. Lightning Login lets internal users log in with Salesforce Authenticator instead of a password. Certain conditions must be met for Lightning Login to succeed.

Call Auth.SessionManagement.getLightningLoginEligibility before or after a login attempt to get the eligibility status. You can call after a login attempt to determine why the login attempt failed.

Enum Values

The following are the values of the Auth.LightningLoginEligibility enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ELIGIBLE</td>
<td>All eligibility conditions are met. The admin has enabled Salesforce Authenticator and Lightning Login, assigned the user Lightning Login user permission, and selected Allow only for users with the Lightning Login User permission from the Session Settings Setup page. The user has set up Salesforce Authenticator and enrolled in Lightning Login.</td>
</tr>
<tr>
<td>ORG.AUTHENTICATOR_NOT_ENABLED</td>
<td>The admin hasn’t enabled Salesforce Authenticator.</td>
</tr>
<tr>
<td>ORG_PREF_NOT_ENABLED</td>
<td>The admin hasn’t enabled Lightning Login. The Admin must select Allow Lightning Login from the Session Settings Setup page.</td>
</tr>
<tr>
<td>USER.AUTHENTICATOR_NOT_CONNECTED</td>
<td>The user hasn’t set up Salesforce Authenticator.</td>
</tr>
<tr>
<td>USER_NOT_ALLOWED</td>
<td>The admin hasn’t granted the user AllowLightningLogin user permission. Allowing Lightning Login to certain users requires the OnlyLLPermUserAllowed org preference. Admins must select Allow only for users with the Lightning Login User permission from the Session Settings Setup page.</td>
</tr>
<tr>
<td>USER_NOT_ENROLLED</td>
<td>The user hasn’t enrolled in Lightning Login.</td>
</tr>
<tr>
<td>USER_PERM_NOT_ENABLED</td>
<td>The admin hasn’t granted the user the Lightning Login Eligible user permission.</td>
</tr>
</tbody>
</table>
LoginDiscoveryHandler Interface

Salesforce gives you the ability to log in users based on other verification methods than username and password. For example, it can prompt users to log in with their email, phone number, or another identifier like a Federation ID or device identifier. Login Discovery is available to these licenses: Customer Community, Customer Community Plus, External Identity, Partner Community, and Partner Community Plus.

Namespace

Auth

Usage

Implement a Auth/LoginDiscoveryHandler for an interview-based log in. The handler looks up a user from the identifier entered, and can call Site.passwordlessLogin to determine which credential to use, such as email or SMS. Or the handler can redirect a user to a third-party identity provider for login. With this handler, the login page doesn’t show a password field. However, you can use Site.passwordlessLogin to then prompt for a password.

From the user perspective, the user enters an identifier at the log in prompt. Then the user completes the login by entering a PIN or password. Or, if SSO-enabled, the user bypasses login.

For an example, see LoginDiscoveryHandler Example Implementation. For more details, see Salesforce Customer Identity in Salesforce Help.

IN THIS SECTION:

LoginDiscoveryHandler Method
LoginDiscoveryHandler Example Implementation

LoginDiscoveryHandler Method

Here’s the method for LoginDiscoveryHandler.

IN THIS SECTION:

login(identifier, startUrl, requestAttributes)

Log in the customer or partner given the specified identifier, such as email or phone number. If successful, redirect the user to the Experience Cloud site page specified by the start URL.

login(identifier, startUrl, requestAttributes)

Log in the customer or partner given the specified identifier, such as email or phone number. If successful, redirect the user to the Experience Cloud site page specified by the start URL.

Signature

public System.PageReference login(String identifier, String startUrl,
Map<String,String>requestAttributes)
Parameters

**identifier**
- **Type:** String
  - Identifier the customer or partner entered at the login prompt, for example, an email address or phone number.

**startUrl**
- **Type:** String
  - Path to the Experience Cloud site page requested by the customer or partner. The user is redirected to this location after successful login.

**requestAttributes**
- **Type:** Map<String,String>
  - Information about the login request based on the user’s browser state when accessing the login page. requestAttributes passes in the CommunityUrl, IPAddress, UserAgent, Platform, Application, City, Country, and Subdivision values. The City, Country, and Subdivision values come from IP geolocation.

Return Value

- **Type:** System.PageReference
  - The URL of the page where the user is redirected.

Example

Here's a sample requestAttributes response.

```
IpAddress=55.555.0.0
UserAgent=Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_4) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/11.1 Safari/605.1.15
Platform=Mac OSX
Application=Browser
City=San Mateo
Country=United States
Subdivision=California
```

LoginDiscoveryHandler Example Implementation

This Apex code example implements the Auth.LoginDiscoveryHandler interface. It checks whether the user who is logging in has a verified email or phone number, depending on which identifier was supplied on the login page. If verified, with Auth.VerificationMethod.EMAIL or Auth.VerificationMethod.SMS, we send a challenge to the identifier, either the user's email address or mobile device. If the user enters the code correctly on the verify page, the user is redirected to the Experience Cloud site's page specified by the start URL. If the user isn't verified, the user must enter a password to log in. The handler also checks that the email and phone number are unique with this code: users.size()==1.

**Note:** Passwordless login works only with verified methods. You can check the verification status on the User object, for example, with User list view, a report, or the API. Make sure that your solution handles the case where the user doesn't have a verification method. This code example falls back to a password.

The default discoverable login handler checks whether the user entered a valid email address or phone number before redirecting the user to the verification page. If an invalid entry is made, the handler returns an error. Because this behavior is vulnerable to user enumeration attack, make sure that your solution prevents this attack. For example, you can create a dummy page similar to
the verification page and redirect the user to the dummy page when invalid user identifier is entered. Also, use generic error messages to avoid providing additional information.

The `discoveryResult` function calls the `Site.passwordlessLogin` method to log the user in with the specified verification method. The `getSsoRedirect` function looks up whether the user logs in with SAML or an Auth Provider. Add the implementation-specific logic to handle the lookup.

global class AutocreatedDiscLoginHandler1535377170343 implements Auth.LoginDiscoveryHandler {

    global PageReference login(String identifier, String startUrl, Map<String, String> requestAttributes) {
        if (identifier != null && isValidEmail(identifier)) {
            // Search for user by email.
            List<User> users = [SELECT Id FROM User WHERE Email = :identifier AND IsActive = TRUE];
            if (!users.isEmpty() && users.size() == 1) {
                // User must have a verified email before using this verification method.
                // We cannot send messages to unverified emails.
                // You can check if the user's email verified bit set and add the password verification method as fallback.
                List<TwoFactorMethodsInfo> verifiedInfo = [SELECT HasUserVerifiedEmailAddress FROM TwoFactorMethodsInfo WHERE UserId = :users[0].Id];
                if (!verifiedInfo.isEmpty() && verifiedInfo[0].HasUserVerifiedEmailAddress == true) {
                    // Use email verification method if the user's email is verified.
                    return discoveryResult(users[0], Auth.VerificationMethod.EMAIL, startUrl, requestAttributes);
                } else {
                    // Use password verification method as fallback
                    // if the user's email is unverified.
                    return discoveryResult(users[0], Auth.VerificationMethod.PASSWORD, startUrl, requestAttributes);
                }
            } else {
                throw new Auth.LoginDiscoveryException('No unique user found. User count=' + users.size());
            }
        }
        if (identifier != null) {
            String formattedSms = getFormattedSms(identifier);
            if (formattedSms != null) {
                // Search for user by SMS.
                List<User> users = [SELECT Id FROM User WHERE MobilePhone = :formattedSms AND IsActive = TRUE];
                if (!users.isEmpty() && users.size() == 1) {
                    // User must have a verified SMS before using this verification method.
                    // We cannot send messages to unverified mobile numbers.
                    // You can check if the user's mobile verified bit is set or add the password verification method as fallback.
                    List<TwoFactorMethodsInfo> verifiedInfo = [SELECT HasUserVerifiedMobileNumber FROM TwoFactorMethodsInfo WHERE UserId = :users[0].Id];
                    if (!verifiedInfo.isEmpty() && verifiedInfo[0].HasUserVerifiedMobileNumber == true) {
                        // Use SMS verification method if the user's mobile is verified.
                        return discoveryResult(users[0], Auth.VerificationMethod.SMS, startUrl, requestAttributes);
                    } else {
                        throw new Auth.LoginDiscoveryException('No unique user found. User count=' + users.size());
                    }
                }
            }
        }
    }

}
// Use SMS verification method if the user's mobile number is verified.
    return discoveryResult(users[0], Auth.VerificationMethod.SMS, startUrl, requestAttributes);
} else {
    // Use password verification method as fallback if the user's
    // mobile number is unverified.
    return discoveryResult(users[0], Auth.VerificationMethod.PASSWORD, startUrl, requestAttributes);
} else {
    throw new Auth>LoginDiscoveryException('No unique user found. User count=' + users.size());
}

if (identifier != null) {
    // You can customize the code to find user via other attributes,
    // such as SSN or Federation ID.
}
    throw new Auth.LoginDiscoveryException('Invalid Identifier');
}

private boolean isValidEmail(String identifier) {
    String emailRegex = '^[a-zA-Z0-9._|\\%#~`=?&/$^*!}{+-\]+@[a-zA-Z0-9.-\]+\.[a-zA-Z]{2,4}$';
    // source: https://www.regular-expressions.info/email.html
    Pattern EmailPattern = Pattern.compile(emailRegex);
    Matcher EmailMatcher = EmailPattern.matcher(identifier);
    if (EmailMatcher.matches()) { return true; }
    else { return false; }
}

private String getFormattedSms(String identifier) {
    // Accept SMS input formats with 1- or 2-digit country code,
    // 3-digit area code, and 7-digit number.
    // You can customize the SMS regex to allow different formats.
    String smsRegex = '^\+\d{1,2}?\d{1,4};#\d{1,2};\d{3};\d{7}$';

    Pattern smsPattern = Pattern.compile(smsRegex);
    Matcher smsMatcher = smsPattern.matcher(identifier);
    if (smsMatcher.matches()) { try {
        // Format user input into the verified SMS format '+xx xxxxxxxx'
        // before DB lookup. If no country code is provided, append
        // US country code +1 for the default.
        String countryCode = smsMatcher.group(1) == null ? '+1' : smsMatcher.group(1);

        return System.UserManagement.formatPhoneNumber(countryCode, smsMatcher.group(2));
    } catch (System.InvalidParameterValueException e) { return null; }
    } else { return null; }
}

private PageReference getSsoRedirect(User user, String startUrl, Map<String, String>
requestAttributes) {
    // You can look up to check whether the user should log in with
    // SAML or an Auth Provider and return the URL to initialize SSO.
    return null;
}

private PageReference discoveryResult(User user, Auth.VerificationMethod method, String
    startUrl, Map<String, String> requestAttributes) {
    // Only users with an External Identity or community license can log in
    // using Site.passwordlessLogin. Use getSsoRedirect to let your org employees
    // log in to an Experience Cloud site.
    PageReference ssoRedirect = getSsoRedirect(user, startUrl, requestAttributes);
    if (ssoRedirect != null) {
        return ssoRedirect;
    } else {
        if (method != null) {
            List<Auth.VerificationMethod> methods = new List<Auth.VerificationMethod>();
            methods.add(method);
            PageReference pwdlessRedirect = Site.passwordlessLogin(user.Id, methods,
            startUrl);
            if (pwdlessRedirect != null) {
                return pwdlessRedirect;
            } else {
                throw new Auth.LoginDiscoveryException('No Passwordless Login redirect URL
returned for verification method: ' + method);
            }
        } else {
            throw new Auth.LoginDiscoveryException('No method found');
        }
    }
}

Code Example: Filter Login Discovery Users by Profile

Your production org can have multiple users with the same verified email address and mobile number. But your customers must have
unique ones. To address this problem, you can add a few lines of code that filters users by profile to ensure uniqueness. This code example
handles users with the External Identity User profile, but can be adapted to support other use cases. For example, you can modify the
first line of code to address users with other user licenses or criteria.

Login Discovery is available with the following user licenses: Customer Community, Customer Community Plus, External Identity, Partner
Community, and Partner Community Plus. It depends on which profiles have access to your Experience Cloud site.

global class AutocreatedDiscLoginHandler1551301979709 implements Auth.LoginDiscoveryHandler
{

global PageReference login(String identifier, String startUrl, Map<String, String>
    requestAttributes) {
    if (identifier != null && isValidEmail(identifier)) {
        // Ensure uniqueness by profile
        Profile p = [SELECT id FROM profile WHERE name = 'External Identity User'];
        List<User> users = [SELECT Id FROM User WHERE Email = :identifier AND IsActive =
TRUE AND profileId=:p.id];
        if (!users.isEmpty() && users.size() == 1) {
            // User must have verified email before using this verification method. We

cannot send messages to unverified emails.
  // You can check if the user has email verified bit on and add the password
verification method as fallback.
  List<TwoFactorMethodsInfo> verifiedInfo = [SELECT HasUserVerifiedEmailAddress
FROM TwoFactorMethodsInfo WHERE UserId = :users[0].Id];
  if (!verifiedInfo.isEmpty() && verifiedInfo[0].HasUserVerifiedEmailAddress ==
true) { // Use email verification method if the user's email is verified.
    return discoveryResult(users[0], Auth.VerificationMethod.EMAIL, startUrl,
requestAttributes);
  } else { // Use password verification method as fallback if the user's email is
unverified.
    return discoveryResult(users[0], Auth.VerificationMethod.PASSWORD, startUrl,
requestAttributes);
  }
} else {
  throw new Auth.LoginDiscoveryException('No unique user found. User count=' +
users.size());
}
}
if (identifier != null) {
  String formattedSms = getFormattedSms(identifier);
  if (formattedSms != null) {
    // Ensure uniqueness by profile
    Profile p = [SELECT id FROM profile WHERE name = 'External Identity User'];
    List<User> users = [SELECT Id FROM User WHERE MobilePhone = :formattedSms AND
IsActive = TRUE AND profileId=:p.id];
    if (!users.isEmpty() && users.size() == 1) { // User must have verified SMS before using this verification method. We
cannot send messages to unverified mobile numbers.
      // You can check if the user has mobile verified bit on or add the password
verification method as fallback.
      List<TwoFactorMethodsInfo> verifiedInfo = [SELECT HasUserVerifiedMobileNumber
FROM TwoFactorMethodsInfo WHERE UserId = :users[0].Id];
      if (!verifiedInfo.isEmpty() && verifiedInfo[0].HasUserVerifiedMobileNumber
== true) { // Use SMS verification method if the user's mobile number is verified.
        return discoveryResult(users[0], Auth.VerificationMethod.SMS, startUrl,
requestAttributes);
      } else { // Use password verification method as fallback if the user's mobile
number is unverified.
        return discoveryResult(users[0], Auth.VerificationMethod.PASSWORD,
startUrl, requestAttributes);
      } else { // User has no unique mobile number. We cannot set SMS verification.
        throw new Auth.LoginDiscoveryException('No unique user found. User count=' +
users.size());
      }
    }
  }
} else {
  throw new Auth.LoginDiscoveryException('No unique user found. User count='};
// You can customize the code to find user via other attributes, such as SSN or Federation ID
} throw new Auth.LoginDiscoveryException('Invalid Identifier');

private boolean isValidEmail(String identifier) {
    String emailRegex = '^[a-zA-Z0-9._|\\%#~`=?&/$^*!}{+-\]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,4}$';
    // source: https://www.regular-expressions.info/email.html
    Pattern EmailPattern = Pattern.compile(emailRegex);
    Matcher EmailMatcher = EmailPattern.matcher(identifier);
    if (EmailMatcher.matches()) { return true; }
    else { return false; }
}

private String getFormattedSms(String identifier) {
    // Accept SMS input formats with 1 or 2 digits country code, 3 digits area code and 7 digits number
    // You can customize the SMS regex to allow different formats
    String smsRegex = '^(\+?\d{1,2}?[\s-])?(\(?\d{3}\)?[\s-]?\d{3}[\s-]?)?\d{4}$';
    Pattern smsPattern = Pattern.compile(smsRegex);
    Matcher smsMatcher = smsPattern.matcher(identifier);
    if (smsMatcher.matches()) {
        try {
            // Format user input into the verified SMS format '+xx xxxxxxx' before DB lookup
            // Append US country code +1 by default if no country code is provided
            String countryCode = smsMatcher.group(1) == null ? '+1' : smsMatcher.group(1);
            return System.UserManagement.formatPhoneNumber(countryCode, smsMatcher.group(2));
        }
        catch(System.InvalidParameterValueException e) {
            return null;
        }
    }
    else { return null; }
}

private PageReference getSsoRedirect(User user, String startUrl, Map<String, String> requestAttributes) {
    // You can look up if the user should log in with SAML or an Auth Provider and return the URL to initialize SSO.
    return null;
}

private PageReference discoveryResult(User user, Auth.VerificationMethod method, String startUrl, Map<String, String> requestAttributes) {
    // Only users with an External Identity or community license can login using Site.passwordlessLogin
    // Use getSsoRedirect to enable your org employees to log in to an Experience Cloud site
    PageReference ssoRedirect = getSsoRedirect(user, startUrl, requestAttributes);
    if (ssoRedirect != null) {

return ssoRedirect;
} else {
    if (method != null) {
        List<Auth.VerificationMethod> methods = new List<Auth.VerificationMethod>();
        methods.add(method);
        PageReference pwdlessRedirect = Site.passwordlessLogin(user.Id, methods,
        startUrl);
        if (pwdlessRedirect != null) {
            return pwdlessRedirect;
        } else {
            throw new Auth.LoginDiscoveryException('No Passwordless Login redirect URL
        returned for verification method: ' + method);
        } else {
            throw new Auth.LoginDiscoveryException('No method found');
        }
    }
}

LoginDiscoveryMethod Enum

Contains methods used to verify the user’s identity when the My Domain login process uses Login Discovery.

Usage

Specifies the verification method used to authenticate internal users when My Domain is set up for Login Discovery.

Enum Values

Auth>LoginDiscoveryMethod enum has the following values.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIGHTNING_LOGIN</td>
<td>Verify identity by Lightning Login, which lets internal users log in with Salesforce Authenticator.</td>
</tr>
<tr>
<td>PASSWORD</td>
<td>Verify identity by entering a password.</td>
</tr>
</tbody>
</table>

MyDomainLoginDiscoveryHandler Interface

The handler used to implement the My Domain Login Discovery page, which is an interview-based (two-step) login process. First the user is prompted for a unique identifier such as an email address or phone number. Then the handler determines (discovers) how to authenticate the user. Either the user enters a password or is directed to an identity provider’s login page.

Namespace

Auth
Usage

Implement `MyDomainLoginDiscoveryHandler` to let My Domain users log in with something other than their username and password. This handler contains the logic to look up the user based on the identifier value entered on the login page. The `Auth.MyDomainLoginDiscoveryHandler.login` method is invoked when the identifier page is submitted and finds the user that corresponds to the submitted identifier. The `Auth.SessionManagement.finishLoginDiscovery` method sends the user to the authentication mechanism and then logs in the user.

Register the handler from the My Domain Setup page. Under Authentication Configuration, select the **Discovery** Login Page Type. For Login Discovery Handler, select this handler from the list of Apex classes.

For an example, see `MyDomainLoginDiscoveryHandler Example Implementation`. For more details, search for My Domain Login Discovery in *Salesforce Help*.

IN THIS SECTION:

- `MyDomainLoginDiscoveryHandler Method`
- `MyDomainLoginDiscoveryHandler Example Implementation`

**MyDomainLoginDiscoveryHandler Method**

`MyDomainLoginDiscoveryHandler` has the following method.

IN THIS SECTION:

- `login(identifier, startUrl, requestAttributes)`

Log in a Salesforce user given the specified identifier, such as email or phone number. If successful, redirect the user to the page specified by the start URL.

**login(identifier, startUrl, requestAttributes)**

Log in a Salesforce user given the specified identifier, such as email or phone number. If successful, redirect the user to the page specified by the start URL.

**Signature**

```java
public System.PageReference login(String identifier, String startUrl, Map<String,String> requestAttributes)
```

**Parameters**

- `identifier`
  - Type: `String`
  - Identifier the Salesforce user entered at the login prompt, for example, an email address or phone number.

- `startUrl`
  - Type: `String`
  - The page users see after successfully logging in to the My Domain subdomain.

- `requestAttributes`
  - Type: `Map <String, String>`
Information about the login request based on the user's browser state when accessing the login page. requestAttributes passes in the MyDomainUrl, IpAddress, UserAgent, Platform, Application, City, Country, and Subdivision values. The City, Country, and Subdivision values come from IP address geolocation.

Return Value
Type: System.PageReference
The URL of the page where the user is redirected to complete authentication.

Example
Here's a sample requestAttributes response.

| CommunityUrl                  | http://my-dev-ed.my.salesforce.com:5555/discover |
| IpAddress                    | 55.255.0.0                                       |
| UserAgent                    | Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_4) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/11.1 Safari/605.1.15 |
| Platform                     | Mac OSX                                          |
| Application                  | Browser                                          |
| City                         | San Mateo                                        |
| Country                      | United States                                    |
| Subdivision                  | California                                       |

MyDomainLoginDiscoveryHandler Example Implementation
Here's an example of the Auth.MyDomainLoginDiscoveryHandler interface. This sample class contains the default logic for My Domain login discovery using password authentication. You can customize the code to ensure it meets your needs. The requestAttributes parameter provides additional information that you can use in the discovery logic. Attributes include MyDomainUrl, IpAddress, UserAgent, and location information (such as Country and City). Use Auth.DiscoveryCustomErrorException to throw custom errors to display on the login page.

To implement this interface, the My Domain login page type must be set to Discovery.

```java
Option-1
```

// This sample class contains the default logic for My Domain login discovery by password.
// You can customize the code to ensure it meets your needs. The requestAttributes parameter
// provides additional information you can use in the discovery logic. Attributes include MyDomainUrl,
// IpAddress, UserAgent, and location information (such as Country and City).
// Use Auth.DiscoveryCustomErrorException to throw custom errors which will be shown on login page.

global class MyDomainDiscLoginDefaultHandler implements Auth.MyDomainLoginDiscoveryHandler {
    global PageReference login(String identifier, String startUrl, Map<String, String> requestAttributes) { 
        if (identifier != null) {
            // Search for user by email
            List<User> users = [SELECT Id FROM User WHERE Email = :identifier AND IsActive = TRUE];
            if (!users.isEmpty() && users.size() == 1) {
                return discoveryResult(users[0], startUrl, requestAttributes);
            } else {
                throw new Auth.LoginDiscoveryException('No unique user found. User count=' + users.size());
            }
        }
        throw new Auth.LoginDiscoveryException('Invalid Identifier');
    }
}
private PageReference getSsoRedirect(User user, String startUrl, Map<String, String> requestAttributes) {
  // You can lookup if the user should login with SAML or an Auth Provider and return the URL to initialize SSO. For example:
  // SamlSsoConfig SSO = [select Id from SamlSsoConfig where DeveloperName='SamlTest' limit 1];

  // To get the URL for a My Domain subdomain, you can pass null in the communityURL parameter.
  // String ssoUrl = Auth.AuthConfiguration.getSamlSsoUrl(null, startUrl, SSO.Id);
  // return new PageReference(ssoUrl);
  return null;
}

private PageReference discoveryResult(User user, String startUrl, Map<String, String> requestAttributes) {

  PageReference ssoRedirect = getSsoRedirect(user, startUrl, requestAttributes);
  if (ssoRedirect != null) {
    return ssoRedirect;
  } else {
    return Auth.SessionManagement.finishLoginDiscovery(Auth.LoginDiscoveryMethod.password, user.Id);
  }
}

Test Class for MyDomainDiscLoginDefaultHandler Class

The following is the test class for MyDomainDiscoveryLoginHandler. For the test to work, your org must have the My Domain login page type set to Discovery.

// Test class for MyDomainDiscLoginDefaultHandler
@isTest
class MyDomainDiscLoginDefaultHandlerTest {
  /* Test Discoverable handler login.
  Create a user with specific email identifier and invoke login.
  Expected : User should be discovered and pagereference should be returned.
  */
  @isTest static void testLogin() {
    // Create user
    String identifierEmail = getUniqueName() + '@test.org';
    createTestUser(identifierEmail);
    Map<String, String> requestAttributes = new Map<String, String>();
    String startUrl = '';
    MyDomainDiscLoginDefaultHandler myDomainDiscLoginDefaultHandler = new MyDomainDiscLoginDefaultHandler();
    // Invoke login method from handler with the email of user created
    PageReference pageReference = myDomainDiscLoginDefaultHandler.login(identifierEmail, startUrl, requestAttributes);
    // Asser page reference is returned
    System.assertNotEquals(null, pageReference, 'Page reference was not returned');
  }
  /* Test Discoverable handler login with invalid (non-existing) user.
  Expected : Auth.LoginDiscoveryException
  */
  @isTest static void testLoginWithInvalidUser() {
    try {
      Map<String, String> requestAttributes = new Map<String, String>();
      // Create user
      String identifierEmail = getUniqueName() + '@test.org';
      createTestUser(identifierEmail);
      Map<String, String> requestAttributes = new Map<String, String>();
      String startUrl = '';
      MyDomainDiscLoginDefaultHandler myDomainDiscLoginDefaultHandler = new MyDomainDiscLoginDefaultHandler();
      // Invoke login method from handler with the email of user created
      PageReference pageReference = myDomainDiscLoginDefaultHandler.login(identifierEmail, startUrl, requestAttributes);
      // Asser page reference is returned
      System.assertNotEquals(null, pageReference, 'Page reference was not returned');
    } catch (Auth.LoginDiscoveryException exception) {
      System.assertEquals('LoginDiscoveryException', exception.getMessage(), 'Expected Auth.LoginDiscoveryException');
    }
  }
}

137
String startUrl = '';  
String uniqueName = getUniqueName();  
String email = uniqueName + '@test.org';  
MyDomainDiscLoginDefaultHandler myDomainDiscLoginDefaultHandler = new MyDomainDiscLoginDefaultHandler();  
// Invoke login method from handler with non-existing user  
myDomainDiscLoginDefaultHandler.login(email, startUrl, requestAttributes);  
}  
catch (Auth.LoginDiscoveryException loginDiscoveryException) {  
  // Assert exception message  
  System.assert(loginDiscoveryException.getMessage().contains("No unique user found"), 'message=' + loginDiscoveryException.getMessage());  
}  
}  
/*  
Generate a random name  
*/  
private static String getUniqueName() {  
  String orgId = UserInfo.getOrganizationId();  
  String dateString = String.valueOf(Datetime.now()).replace(' ', '').replace(':', '').replace('-', '');  
  Integer randomInt = Integer.valueOf(Math.rint(Math.random() * 1000000));  
  String uniqueName = orgId + dateString + randomInt;  
  return uniqueName;  
}  
/*  
Create user with given email.  
*/  
private static void createTestUser(String identifierEmail) {  
  String uniqueName = getUniqueName();  
  Profile pf = [SELECT Id FROM Profile WHERE Name='Standard User'];  
  String profileID = pf.Id;  
  String fName = 'fname';  
  String lName = uniqueName + '-lname';  
  User tuser = new User(  
    firstname = fName,  
    lastName = lName,  
    email = identifierEmail,  
    Username = uniqueName + '@test.org',  
    EmailEncodingKey = 'ISO-8859-1',  
    Alias = uniqueName.substring(18, 23),  
    TimeZoneSidKey = 'America/Los_Angeles',  
    LocaleSidKey = 'en_US',  
    LanguageLocaleKey = 'en_US',  
    ProfileId = profileID);  
  
  insert tuser;  
}  

OAuthRefreshResult Class

Stores the result of an AuthProviderPluginClass refresh method. OAuth authentication flow provides a refresh token that can be used to get a new access token. Access tokens have a limited lifetime as specified by the session timeout value. When an access token expires, use a refresh token to get a new access token.
Namespace
Auth

Usage

The OAuthRefreshResult class contains the parameters, accessToken, refreshToken, and error, all of which are of type string. For a code example, see .

IN THIS SECTION:

OAuthRefreshResult Constructors
OAuthRefreshResult Properties

OAuthRefreshResult Constructors

The following are constructors for OAuthRefreshResult.

IN THIS SECTION:

OAuthRefreshResult(accessToken, refreshToken, error)
Creates an instance of the OAuthRefreshResult class using the specified access token, refresh token, and error for a custom authentication provider plug-in.

OAuthRefreshResult(accessToken, refreshToken)
Creates an instance of the OAuthRefreshResult class using the specified access token and refresh token for a custom authentication provider plug-in. Use this method when you know that the refresh was successful.

OAuthRefreshResult (accessToken, refreshToken, error)

Creates an instance of the OAuthRefreshResult class using the specified access token, refresh token, and error for a custom authentication provider plug-in.

Signature

class OAuthRefreshResult

public OAuthRefreshResult(String accessToken, String refreshToken, String error)

Parameters

accessToken
Type: String
OAuth access token for the user who is currently logged in.

refreshToken
Type: String
OAuth refresh token for the user who is currently logged in.

error
Type: String
Error that occurred when a user attempted to authenticate with the custom authentication provider.
OAuthRefreshResult(accessToken, refreshToken)

Creates an instance of the OAuthRefreshResult class using the specified access token and refresh token for a custom authentication provider plug-in. Use this method when you know that the refresh was successful.

Signature

public OAuthRefreshResult(String accessToken, String refreshToken)

Parameters

accessToken
Type: String
The OAuth access token for the user who is logged in.

refreshToken
Type: String
The OAuth refresh token for the user who is logged in.

OAuthRefreshResult Properties

The following are properties for OAuthRefreshResult.

IN THIS SECTION:

accessToken
The OAuth access token for the user who is currently logged in.

error
Error that occurs when a user unsuccessfully attempts to authenticate with the custom authentication provider.

refreshToken
The OAuth refresh token for the user who is currently logged in.

accessToken
The OAuth access token for the user who is currently logged in.

Signature

public String accessToken {get; set;}

Property Value

Type: String

error

Error that occurs when a user unsuccessfully attempts to authenticate with the custom authentication provider.
**Signature**

```
public String error {get; set;}
```

**Property Value**

Type: String

**refreshToken**

The OAuth refresh token for the user who is currently logged in.

**Signature**

```
public String refreshToken {get; set;}
```

**Property Value**

Type: String

---

**RegistrationHandler Interface**

Salesforce provides the ability to use an authentication provider, such as Facebook© or Janrain©, for single sign-on into Salesforce.

**Namespace**

Auth

**Usage**

To set up single sign-on, you must create a class that implements `Auth.RegistrationHandler`. Classes implementing the `Auth.RegistrationHandler` interface are specified as the Registration Handler in authentication provider definitions, and enable single sign-on into Salesforce portals and organizations from third-party services such as Facebook. Using information from the authentication providers, your class must perform the logic of creating and updating user data as appropriate, including any associated account and contact records.

**Note:** During the user update process, you can use the `confirmUser()` method to ensure that users are correctly mapped between Salesforce and the third party. For more information, see the ConfirmUserRegistrationHandler Interface.

---

**IN THIS SECTION:**

- RegistrationHandler Methods
- Storing User Information and Getting Access Tokens
- Auth.RegistrationHandler Example Implementation
- Auth.RegistrationHandler Error Example

This example implements the `Auth.RegistrationHandler` interface and shows how to use a custom exception to display an error message in the URL of the page. If you don't use a custom exception, the error code and description appear in the URL and the error description appears on the page.
RegistrationHandler Methods

The following are methods for RegistrationHandler.

IN THIS SECTION:

createUser(portalId, userData)
Returns a User object using the specified portal ID and user information from the third party, such as the username and email address. The User object corresponds to the third party’s user information. It can be a new user that hasn’t been inserted in your org’s database, or it can represent an existing user record in the database. If it’s a new User object, Salesforce inserts a user record for you.

updateUser(userId, portalId, userData)
Updates the specified user’s information. This method is called if the user has logged in before with the authentication provider and then logs in again.

createUser(portalId, userData)
Returns a User object using the specified portal ID and user information from the third party, such as the username and email address. The User object corresponds to the third party’s user information. It can be a new user that hasn’t been inserted in your org’s database, or it can represent an existing user record in the database. If it’s a new User object, Salesforce inserts a user record for you.

Signature

public User createUser(ID portalId, Auth.UserData userData)

Parameters

portalId
Type: ID

userData
Type: Auth.UserData

Return Value

Type: User

Usage

The portalID value can be null or an empty key if there’s no portal configured with this provider.

updateUser(userId, portalId, userData)
Updates the specified user’s information. This method is called if the user has logged in before with the authentication provider and then logs in again.

Signature

public Void updateUser(ID userId, ID portalId, Auth.UserData userData)
Parameters

userId
  Type: ID
portalId
  Type: ID
userData
  Type: Auth.UserData

Return Value
Type: Void

Usage
The portalID value can be null or an empty key if there's no portal configured with this provider.

Storing User Information and Getting Access Tokens

The Auth.UserData class is used to store user information for Auth.RegistrationHandler. The third-party authentication provider can send back a large collection of data about the user, including their username, email address, locale, and so on. Frequently used data is converted into a common format with the Auth.UserData class and sent to the registration handler.

If the registration handler wants to use the rest of the data, the Auth.UserData class has an attributeMap variable. The attribute map is a map of strings (Map<String, String>) for the raw values of all the data from the third party. Because the map is <String, String>, values that the third party returns that aren't strings (like an array of URLs or a map) are converted into an appropriate string representation. The map includes everything returned by the third-party authentication provider, including the items automatically converted into the common format.

The constructor for Auth.UserData has the following syntax:

```java
Auth.UserData(String identifier,
  String firstName,
  String lastName,
  String fullName,
  String email,
  String link,
  String userName,
  String locale,
  String provider,
  String siteLoginUrl,
  Map<String, String> attributeMap)
```

To learn about Auth.UserData properties, see Auth.UserData Class.

⚠️ Note: You can only perform DML operations on additional sObjects in the same transaction with User objects under certain circumstances. For more information, see sObjects That Cannot Be Used Together in DML Operations.

For all authentication providers except Janrain, after a user is authenticated using a provider, the access token associated with that provider for this user can be obtained in Apex using the Auth.AuthToken Apex class. Auth.AuthToken provides two methods to retrieve access tokens. One is `getAccessToken`, which obtains a single access token. Use this method if the user ID is mapped to a single third-party user. If the user ID is mapped to multiple third-party users, use `getAccessTokenMap`, which returns a map
of access tokens for each third-party user. For more information about authentication providers, see Authentication Providers in Salesforce Help.

When using Janrain as an authentication provider, you must use the Janrain `accessCredentials` dictionary values to retrieve the access token or its equivalent. Only some providers supported by Janrain provide an access token, while other providers use other fields. The Janrain `accessCredentials` fields are returned in the `attributeMap` variable of the `Auth.UserData` class. See the Janrain `auth_info` documentation for more information on `accessCredentials`.

![Note: Not all Janrain account types return `accessCredentials`. Sometimes you must change your account type to receive the information.](image)

To learn about the `AuthToken` methods, see `AuthToken Class`.

**Auth.RegistrationHandler Example Implementation**

This example implements the `Auth.RegistrationHandler` interface that creates as well as updates a standard user based on data provided by the authentication provider. Error checking has been omitted to keep the example simple.

```apex
global class StandardUserRegistrationHandler implements Auth.RegistrationHandler{
    global User createUser(Id portalId, Auth.UserData data){
        User u = new User();
        Profile p = [SELECT Id FROM profile WHERE name='Standard User'];
        u.username = data.username + '@salesforce.com';
        u.email = data.email;
        u.lastName = data.lastName;
        u.firstName = data.firstName;
        String alias = data.username;
        if(alias.length() > 8) {
            alias = alias.substring(0, 8);
        }
        u.alias = alias;
        u.languageLocaleKey = data.attributeMap.get('language');
        u.localeSidKey = data.locale;
        u.emailEncodingKey = 'UTF-8';
        u.timeZoneSidKey = 'America/Los_Angeles';
        u.profileId = p.Id;
        return u;
    }

    global void updateUser(Id userId, Id portalId, Auth.UserData data){
        User u = new User(id=userId);
        u.username = data.username + '@salesforce.com';
        u.email = data.email;
        u.lastName = data.lastName;
        u.firstName = data.firstName;
        String alias = data.username;
        if(alias.length() > 8) {
            alias = alias.substring(0, 8);
        }
        u.alias = alias;
        u.languageLocaleKey = data.attributeMap.get('language');
        u.localeSidKey = data.locale;
        update(u);
    }
}
```
The following example tests the above code.

```java
@isTest
private class StandardUserRegistrationHandlerTest {
    static testMethod void testCreateAndUpdateUser() {
        StandardUserRegistrationHandler handler = new StandardUserRegistrationHandler();
        Auth.UserData sampleData = new Auth.UserData('testId', 'testFirst', 'testLast', 'testFirst testLast', 'testuser@example.org', null, 'testuserlong', 'en_US', 'facebook', null, new Map<String, String>{'language' => 'en_US'});
        User u = handler.createUser(null, sampleData);
        System.assertEquals('testuserlong@salesforce.com', u.userName);
        System.assertEquals('testuser@example.org', u.email);
        System.assertEquals('testLast', u.lastName);
        System.assertEquals('testFirst', u.firstName);
        System.assertEquals('testuser', u.alias);
        insert(u);
        String uid = u.id;

        sampleData = new Auth.UserData('testNewId', 'testNewFirst', 'testNewLast', 'testNewFirst testNewLast', 'testnewuser@example.org', null, 'testnewuserlong', 'en_US', 'facebook', null, new Map<String, String>{});
        handler.updateUser(uid, null, sampleData);
        User updatedUser = [SELECT userName, email, firstName, lastName, alias FROM user WHERE id=:uid];
        System.assertEquals('testnewuserlong@salesforce.com', updatedUser.userName);
        System.assertEquals('testnewuser@example.org', updatedUser.email);
        System.assertEquals('testNewLast', updatedUser.lastName);
        System.assertEquals('testNewFirst', updatedUser.firstName);
        System.assertEquals('testnewu', updatedUser.alias);
    }
}
```

**Auth.RegistrationHandler Error Example**

This example implements the `Auth.RegistrationHandler` interface and shows how to use a custom exception to display an error message in the URL of the page. If you don’t use a custom exception, the error code and description appear in the URL and the error description appears on the page.

To limit this example to the custom exception, some code was omitted.

```java
global class RegHandler implements Auth.RegistrationHandler {
    class RegHandlerException extends Exception {

        global User createUser(Id portalId, Auth.UserData data){
            List<Profile> profiles = [SELECT Id, Name, UserType FROM Profile WHERE Name = 'Power User'];
            Profile profile = profiles.isEmpty() ? null : profiles[0];
            if(profile==null)
                throw new RegHandlerException('Cannot find the profile. For help, contact your administrator.');</n        }
    }
```
global void updateUser(Id userId, Id portalId, Auth.UserData data){
    User u = new User(id=userId);
    u.lastName = data.lastName;
    u.firstName = data.firstName;
    update(u);
}

SamlJitHandler Interface
Use this interface to control and customize Just-in-Time user provisioning logic during SAML single sign-on.

Namespace
Auth

Usage
To use custom logic for user provisioning during SAML single sign-on, you must create a class that implements Auth.SamlJitHandler. This allows you to incorporate organization-specific logic (such as populating custom fields) when users log in to Salesforce with single sign-on. Keep in mind that your class must perform the logic of creating and updating user data as appropriate, including any associated account and contact records.

In Salesforce, you specify your class that implements this interface in the SAML JIT Handler field in SAML Single Sign-On Settings. Make sure that the user you specify to run the class has "Manage Users" permission.

IN THIS SECTION:
   SamlJitHandler Methods
   SamlJitHandler Example Implementation

SamlJitHandler Methods
The following are methods for SamlJitHandler.

IN THIS SECTION:
   createUser(samlSsoProviderId, communityId, portalId, federationId, attributes, assertion)
   Returns a User object using the specified Federation ID. The User object corresponds to the user information. This object can be a new user that hasn’t been inserted in the database or an existing user record in the database.

   updateUser(userId, samlSsoProviderId, communityId, portalId, federationId, attributes, assertion)
   Updates the specified user’s information. This method is called if the user has logged in before with SAML single sign-on and then logs in again, or if your application is using the Existing User Linking URL.
**createUser**(*samlSsoProviderId*, *communityId*, *portalId*, *federationId*, *attributes*, *assertion*)

Returns a User object using the specified Federation ID. The User object corresponds to the user information. This object can be a new user that hasn’t been inserted in the database or an existing user record in the database.

**Signature**

```java
public User createUser(Id samlSsoProviderId, Id communityId, Id portalId, String federationId, Map<String,String> attributes, String assertion)
```

**Parameters**

- **samlSsoProviderId**
  - Type: Id
  - The ID of the SamlSsoConfig standard object.

- **communityId**
  - Type: Id
  - The ID of the Experience Cloud site. This parameter can be `null` if you’re not creating an Experience Cloud user.

- **portalId**
  - Type: Id
  - The ID of the portal. This parameter can be `null` if you’re not creating a portal user.

- **federationId**
  - Type: String
  - The ID Salesforce expects to be used for this user.

- **attributes**
  - Type: `Map<String,String>`
  - All attributes in the SAML assertion that were added to the default assertion; for example, custom attributes. Attributes are case-sensitive.
  - If the assertion is encrypted, the attribute map contains a decrypted assertion stored as a value with the key `Sfdc.SamlAssertion`.

- **assertion**
  - Type: String
  - The default SAML assertion, base-64 encoded.
  - If the assertion is encrypted, this parameter is also encrypted. To access the decrypted assertion, see the `Sfdc.SamlAssertion` key in the attribute map.

**Return Value**

- Type: User
- A User sObject.
Usage

The `communityId` and `portalId` parameter values can be `null` or the associated keys can be empty if there's no Experience Cloud site or portal configured with this organization.

**updateUser(userId, samlSsoProviderId, communityId, portalId, federationId, attributes, assertion)**

Updates the specified user's information. This method is called if the user has logged in before with SAML single sign-on and then logs in again, or if your application is using the Existing User Linking URL.

Signature

```java
public void updateUser(Id userId, Id samlSsoProviderId, Id communityId, Id portalId,
String federationId, Map<String,String> attributes, String assertion)
```

Parameters

- `userId`
  Type: `Id`
  The ID of the Salesforce user.

- `samlSsoProviderId`
  Type: `Id`
  The ID of the SamlSsoConfig object.

- `communityId`
  Type: `Id`
  The ID of the Experience Cloud site. This type can be `null` if you're not updating an Experience Cloud user.

- `portalId`
  Type: `Id`
  The ID of the portal. This type can be `null` if you're not updating a portal user.

- `federationId`
  Type: `String`
  The ID Salesforce expects to be used for this user.

- `attributes`
  Type: `Map<String,String>`
  All attributes in the SAML assertion that were added to the default assertion; for example, custom attributes. Attributes are case-sensitive.
  If the assertion is encrypted, the attribute map also contains a decrypted assertion stored as a value with the key `Sfdc.SamlAssertion`.

- `assertion`
  Type: `String`
  The default SAML assertion, base-64 encoded.
  If the assertion is encrypted, this parameter is also encrypted. To access the decrypted assertion, see the `Sfdc.SamlAssertion` key in the attribute map.
SamlJitHandler Example Implementation

This is an example implementation of the Auth.SamlJitHandler interface. This code uses private methods to handle accounts and contacts (handleContact() and handleAccount()), which aren’t included in this example.

global class StandardUserHandler implements Auth.SamlJitHandler {
    private class JitException extends Exception{}
    private void handleUser(boolean create, User u, Map<String, String> attributes,
                            String federationIdentifier, boolean isStandard) {
        if(create && attributes.containsKey('User.Username')) {
            u.Username = attributes.get('User.Username');
        }
        if(create) {
            if(attributes.containsKey('User.FederationIdentifier')) {
                u.FederationIdentifier = attributes.get('User.FederationIdentifier');
            } else {
                u.FederationIdentifier = federationIdentifier;
            }
        }
        if(attributes.containsKey('User.ProfileId')) {
            String profileId = attributes.get('User.ProfileId');
            Profile p = [SELECT Id FROM Profile WHERE Id=:profileId];
            u.ProfileId = p.Id;
        }
        if(attributes.containsKey('User.UserRoleId')) {
            String userRole = attributes.get('User.UserRoleId');
            UserRole r = [SELECT Id FROM UserRole WHERE Id=:userRole];
            u.UserRoleId = r.Id;
        }
        if(attributes.containsKey('User.Phone')) {
            u.Phone = attributes.get('User.Phone');
        }
        if(attributes.containsKey('User.Email')) {
            u.Email = attributes.get('User.Email');
        }
    }
    private void handleJit(boolean create, User u, Id samlSsoProviderId, Id communityId,
                           Id portalId,
                           String federationIdentifier, Map<String, String> attributes, String assertion) {
        if(communityId != null || portalId != null) {
            String account = handleAccount(create, u, attributes);
            //More attributes here - removed for length
            //Handle custom fields here
            if(!create) {
                update(u);
            }
        }
    }
}
handleContact(create, account, u, attributes);
handleUser(create, u, attributes, federationIdentifier, false);
} else {
    handleUser(create, u, attributes, federationIdentifier, true);
}
}

global User createUser(Id samlSsoProviderId, Id communityId, Id portalId, String federationIdentifier, Map<String, String> attributes, String assertion) {
    User u = new User();
    handleJit(true, u, samlSsoProviderId, communityId, portalId, federationIdentifier, attributes, assertion);
    return u;
}

global void updateUser(Id userId, Id samlSsoProviderId, Id communityId, Id portalId, String federationIdentifier, Map<String, String> attributes, String assertion) {
    User u = [SELECT Id FROM User WHERE Id=:userId];
    handleJit(false, u, samlSsoProviderId, communityId, portalId, federationIdentifier, attributes, assertion);
}

# SessionManagement Class

Contains methods for verifying users’ identity, creating custom login flows, customizing security levels, and defining trusted IP ranges for a current session.

## Namespace

Auth

## SessionManagement Methods

The following are methods for SessionManagement. All methods are static. Use these methods to customize your user identity verification flows, manage the use of time-based one-time password (TOTP) apps like Google Authenticator, or create custom login flows. Other methods validate a user’s incoming IP address against trusted IP range settings for an organization or profile.

### IN THIS SECTION:

- finishLoginDiscovery(method, userId)
  Finishes the My Domain Login Discovery login process.
- finishLoginFlow()
  Finish the Visualforce Page login flow process, and redirect the user to the default home page.
- finishLoginFlow(startUrl)
  Finish the Visualforce Page login flow process, and redirect the user to the specified start URL.
generateVerificationUrl(policy, description, destinationUrl)
Initiates a user identity verification flow with the verification method that the user registered with, and returns a URL to the identity verification screen. For example, if you have a custom Visualforce page that displays sensitive account details, you can prompt the user to verify identity before viewing it.

generateVerificationUrl

getCurrentSession()
Returns a map of attributes for the current session.

generateVerificationUrl

getLightningLoginEligibility(userId)
Returns the eligibility status for a user who’s logging in with Lightning Login when you set up your org with My Domain and use the Login Discovery page type. Use this method to redirect the user to a custom login flow. For example, use after a login attempt to redirect the user to password flow if the user is ineligible for Lightning Login.

generateVerificationUrl

getQrCode()
Returns a map containing a URL to a quick response (QR) code and a time-based one-time password (TOTP) shared secret to configure authenticator apps or devices for multi-factor authentication (MFA).

generateVerificationUrl

getRequiredSessionLevelForProfile(profileId)
Indicates the required login security session level for the given profile.

generateVerificationUrl

ignoreForConcurrentSessionLimit(sessions)
This method is reserved for internal Salesforce use.

generateVerificationUrl

inOrgNetworkRange(ipAddress)
Indicates whether the given IP address is within the organization’s trusted IP range according to the organization’s Network Access settings.

generateVerificationUrl

isIpAllowedForProfile(profileId, ipAddress)
Indicates whether the given IP address is within the trusted IP range for the given profile.

generateVerificationUrl

setSessionLevel(level)
Sets the user’s current session security level.

generateVerificationUrl

validateTotpTokenForKey(sharedKey, totpCode)
Deprecated. Use validateTotpTokenForKey(totpSharedKey, totpCode, description) instead.

generateVerificationUrl

validateTotpTokenForUser(totpCode)
Deprecated. Use validateTotpTokenForUser(totpCode, description) instead.

generateVerificationUrl

validateTotpTokenForUser(totpCode, description)
Indicates whether a time-based one-time password (TOTP) code (token) is valid for the current user.

generateVerificationUrl

verifyDeviceFlow(userId, startUrl)
Verifies the user code entered during the device authentication flow and redirects users to the OAuth approval page. If users aren’t logged in, they must log in. After successful login, users are prompted to allow the device to access Salesforce data.

generateVerificationUrl

finishLoginDiscovery(method, userId)
Finishes the My Domain Login Discovery login process.
Signature

```java
public static System.PageReference finishLoginDiscovery(Auth.LoginDiscoveryMethod method, Id userId)
```

Parameters

```java
method
Type: Auth.LoginDiscoveryMethod LoginDiscoveryMethod Enum
Verification method used with My Domain Login Discovery.
```

```java
userId
Type: Id
ID used to log in the user. The user must be active.
```

Return Value

Type: System.PageReference

Usage

Include this method when implementing the `MyDomainLoginDiscoveryHandler` interface to direct users to an authentication mechanism, and then log them in. If users enter a username in the login page, they are sent to the password page for authentication. If users are enrolled in Lightning Login, they are directed to the Salesforce Authenticator to authenticate. If users are SSO-enabled, they are sent to the suitable identity provider (IdP) to authenticate.

The calling user requires Manage Users permission. If the user passed in is frozen or inactive, the method throws an exception.

After implementing the `MyDomainLoginDiscoveryHandler` interface, register the Login Discovery handler from the My Domain Setup page. Under Authentication Configuration, select this handler from the list of Apex classes.

### finishLoginFlow()

Finish the Visualforce Page login flow process, and redirect the user to the default home page.

Signature

```java
public static System.PageReference finishLoginFlow()
```

Return Value

Type: System.PageReference

Usage

Include this method in the Apex controller of the Visualforce Page login flow when creating login flows programmatically. This method indicates that the login flow is finished and redirects the user to the Experience Cloud site’s default home page. The login process runs in a restricted session until users complete the process. Calling this method indicates that the login flow is complete, lifts the restriction, and gives users full access to the Experience Cloud site.
**finishLoginFlow(startUrl)**

Finish the Visualforce Page login flow process, and redirect the user to the specified start URL.

**Signature**

```java
public static System.PageReference finishLoginFlow(String startUrl)
```

**Parameters**

- `startUrl`
  - Type: String
  - Path to the page that users see when they access the Experience Cloud site.

**Return Value**

- Type: `System.PageReference`

**Usage**

Include this method in the Apex controller of the Visualforce Page login flow when creating login flows programmatically. This method indicates that the login flow is finished and redirects the user to the specified location in the Experience Cloud site. The login process runs in a restricted session until users complete the process. Calling this method indicates that the login flow is complete, lifts the restriction, and gives users full access to the Experience Cloud site.

**generateVerificationUrl(policy, description, destinationUrl)**

Initiates a user identity verification flow with the verification method that the user registered with, and returns a URL to the identity verification screen. For example, if you have a custom Visualforce page that displays sensitive account details, you can prompt the user to verify identity before viewing it.

**Signature**

```java
public static String generateVerificationUrl(Auth.VerificationPolicy policy, String description, String destinationUrl)
```

**Parameters**

- `policy`
  - Type: `Auth.VerificationPolicy`
  - The session security policy required to initiate identity verification for the user’s session. For example, if the policy is set to High Assurance level of session security, and the user’s current session has the standard level of session security, the user’s session is raised to high assurance after successful verification of identity. In the Setup user interface, this value is shown in the Triggered By column of Identity Verification History.

- `description`
  - Type: String
  - The custom description that describes the activity requiring identity verification; for example, “Complete purchase and check out”. This text appears to users when they verify their identity in Salesforce and, if they use Salesforce Authenticator version 2 or later, in
the Salesforce Authenticator mobile app. In addition, in the Setup user interface, this text is shown in the Activity Message column of Identity Verification History.

**destinationUrl**

Type: **String**

The relative or absolute Salesforce URL that you want to redirect the user to after identity verification—for example, /apex/mypage. The user is redirected to `destinationUrl` when the identity verification flow is complete, regardless of success. For example, if a user chooses to not respond to the identity challenge and cancels it, the user is still redirected to `destinationUrl`. As a best practice, ensure that your code for this page manually checks that the security policy was satisfied (and the user didn’t just manually type the URL in the browser). For example, if the `policy` is High Assurance, the target page checks that the user’s session is high assurance before allowing access.

**Return Value**

Type: **String**

The URL where the user is redirected to verify identity.

**Usage**

- If the user is already registered to confirm identity using a time-based one-time password (TOTP), then the user is redirected to the one-time password identity verification flow and asked to provide a code.
- If the user isn’t registered with any verification method (such as one-time password or Salesforce Authenticator version 2 or later), the user is prompted to download and verify identity using Salesforce Authenticator. The user can also choose a different verification method.

**getcurrentSession()**

Returns a map of attributes for the current session.

**Signature**

```java
public static Map<String, String> getCurrentSession()
```

**Return Value**

Type: **Map<String, String>**

**Usage**

The map includes a `ParentId` value, which is the 18-character ID for the parent session, if one exists (for example, if the current session is for a canvas app). If the current session doesn’t have a parent, this value is null. The map also includes the `LogoutUrl` assigned to the current session.

If you create an Apex test method that calls this method, the test fails with an error such as, “Unexpected Exception: Current session unavailable.” An error occurs because there isn’t a session in the context through which the test is being run.

When a session is reused, Salesforce updates the `LoginHistoryId` with the value from the most recent login.
The following example shows the name-value pairs in a map returned by `getCurrentSession()`. Note that `UsersId` includes an "s" in the name to match the name of the corresponding field in the AuthSession object.

```java
{
    SessionId=0Ak###############,
    UserType=Standard,
    ParentId=0Ak###############,
    NumSecondsValid=7200,
    LoginType=SAML Idp Initiated SSO,
    LoginDomain=null,
    LoginHistoryId=0Ya###############,
    Username=user@domain.com,
    CreatedDate=Wed Jul 30 19:09:29 GMT 2014,
    SessionType=Visualforce,
    LastModifiedDate=Wed Jul 30 19:09:16 GMT 2014,
    LogoutUrl=https://google.com,
    SessionSecurityLevel=STANDARD,
    UsersId=005###############,
    SourceIp=1.1.1.1
}
```

`getLightningLoginEligibility(userId)`

Returns the eligibility status for a user who’s logging in with Lightning Login when you set up your org with My Domain and use the Login Discovery page type. Use this method to redirect the user to a custom login flow. For example, use after a login attempt to redirect the user to password flow if the user is ineligible for Lightning Login.

**Signature**

```java
public static Auth.LightningLoginEligibility getLightningLoginEligibility(Id userId)
```

**Parameters**

- `userId`
  - Type: `Id`
  - ID of the user who is logging in.

**Return Value**

- Type: `Auth.LightningLoginEligibility`
  - Returns the current eligibility status.

**Example**

```java
Auth.LightningLoginEligibility eligibility = Auth.SessionManagement.getLightningLoginEligibility(id);
if (eligibility == Auth.LightningLoginEligibility.ELIGIBLE) {
    // success
}
```
**getQrCode()**

Returns a map containing a URL to a quick response (QR) code and a time-based one-time password (TOTP) shared secret to configure authenticator apps or devices for multi-factor authentication (MFA).

**Signature**

```java
public static Map<String, String> getQrCode()
```

**Return Value**

Type: `Map<String, String>`

**Usage**

The QR code encodes the returned secret as well as the current user's username. The keys are `qrCodeUrl` and `secret`. Calling this method does not change any state for the user, nor does it read any state from the user. This method returns a brand new secret every time it is called, does not save that secret anywhere, and does not validate the TOTP token. The admin must explicitly save the values for the user after verifying a TOTP token with the secret.

The secret is a base32-encoded string of a 20-byte shared key.

**Example**

The following is an example of how to request the QR code.

```java
public String getGetQRCode() {
    return getQrCode();
}
public String getQRCode() {
    Map<String, String> codeResult = Auth.SessionManagement.getQrCode();
    String result = 'URL: ' + codeResult.get('qrCodeUrl') + ' SECRRT: ' + codeResult.get('secret');
    return result;
}
```

The following is an example of a returned map.

```json
{qrCodeUrl=https://www.salesforce.com/secur/qrCode?w=200&h=200&t=tf&u=user%0000000000.com&s=AAAAA7B5BBBB5AAAAAAA66BBBB,
secret=AAAAA7B5AAAAAA5BBBBBBBB66AAA}
```

**getRequiredSessionLevelForProfile(profileId)**

Indicates the required login security session level for the given profile.

**Signature**

```java
public static Auth.SessionLevel getRequiredSessionLevelForProfile(String profileId)
```

**Parameters**

- `profileId`
  
  Type: `String`
The 15-character profile ID.

Return Value
Type: Auth.SessionLevel
The session security level required at login for the profile with the ID profileId. You can customize the assignment of each level in Session Settings. For example, you can set the High Assurance level to apply only to users who authenticated with multi-factor authentication (MFA) or through a specific identity provider.

ignoreForConcurrentSessionLimit(sessions)
This method is reserved for internal Salesforce use.

Signature
public static Map<String, String> ignoreForConcurrentSessionLimit(Object sessions)

Parameters
sessions
Type: Object

Return Value
Type: Map<String, String>

inOrgNetworkRange(ipAddress)
Indicates whether the given IP address is within the organization's trusted IP range according to the organization's Network Access settings.

Signature
public static Boolean inOrgNetworkRange(String ipAddress)

Parameters
ipAddress
Type: String
The IP address to validate.

Return Value
Type: Boolean

Usage
If a trusted IP range is not defined, this returns false, and throws an exception if the IP address is not valid.
<table>
<thead>
<tr>
<th>Trusted IP Range Exists?</th>
<th>User is in the Trusted IP Range?</th>
<th>Return Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>Yes</td>
<td>true</td>
</tr>
<tr>
<td>Yes</td>
<td>No</td>
<td>false</td>
</tr>
<tr>
<td>No</td>
<td>N/A</td>
<td>false</td>
</tr>
</tbody>
</table>

**isIpAllowedForProfile(profileId, ipAddress)**

Indicates whether the given IP address is within the trusted IP range for the given profile.

**Signature**

```java
public static Boolean isIpAllowedForProfile(String profileId, String ipAddress)
```

**Parameters**

- **profileId**
  - Type: `String`
  - The 15-character alphanumeric string for the current user’s profile ID.

- **ipAddress**
  - Type: `String`
  - The IP address to validate.

**Return Value**

Type: `Boolean`

**Usage**

If a trusted IP range is not defined, this returns `true`, and throws an exception if the IP address is not valid or if the profile ID is not valid.

<table>
<thead>
<tr>
<th>Trusted IP Range Exists?</th>
<th>User is in the Trusted IP Range?</th>
<th>Return Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>Yes</td>
<td>true</td>
</tr>
<tr>
<td>Yes</td>
<td>No</td>
<td>false</td>
</tr>
<tr>
<td>No</td>
<td>N/A</td>
<td>true</td>
</tr>
</tbody>
</table>

**setSessionLevel(level)**

Sets the user’s current session security level.

**Signature**

```java
public static Void setSessionLevel(Auth.SessionLevel level)
```
Parameters

level
Type: Auth.SessionLevel

The session security level to assign to the user. The meaning of each level can be customized in the Session Settings for each organization, such as setting the High Assurance level to apply only to users who authenticated with multi-factor authentication (MFA) or through a specific identity provider.

Return Value

Type: Void

Usage

This setting affects the session level of all sessions associated with the current session, such as Visualforce or UI access.

If you create an Apex test method that calls this method, the test fails with an error such as, "Unexpected Exception: Current session unavailable." An error occurs because there isn’t a session in the context through which the test is being run.

Example

The following is an example class for setting the session level.

```apex
public class RaiseSessionLevel{
    public void setLevelHigh() {
        Auth.SessionManagement.setSessionLevel(Auth.SessionLevel.HIGH_ASSURANCE);
    }
    public void setLevelStandard() {
        Auth.SessionManagement.setSessionLevel(Auth.SessionLevel.STANDARD);
    }
}
```

validateTotpTokenForKey(sharedKey, totpCode)

Deprecated. Use validateTotpTokenForKey(totpSharedKey, totpCode, description) instead.

Signature

public static Boolean validateTotpTokenForKey(String sharedKey, String totpCode)

Parameters

sharedKey
Type: String

The shared (secret) key. The sharedKey must be a base32-encoded string of a 20-byte shared key.

totpCode
Type: String

The time-based one-time password (TOTP) code to validate.
Return Value
Type: Boolean

Usage
If the key is invalid or doesn’t exist, this method throws an invalid parameter value exception or a no data found exception, respectively. If the current user exceeds the maximum of 10 token validation attempts, this method throws a security exception.

**validateTotpTokenForKey(totpSharedKey, totpCode, description)**
Indicates whether a time-based one-time password (TOTP) code (token) is valid for the given shared key.

**Signature**

```java
public static Boolean validateTotpTokenForKey(String totpSharedKey, String totpCode, String description)
```

**Parameters**

totpSharedKey
Type: String
The shared (secret) key. The `totpSharedKey` must be a base32-encoded string of a 20-byte shared key.

totpCode
Type: String
The time-based one-time password (TOTP) code to validate.

description
Type: String
The custom description that describes the activity requiring identity verification; for example, “Complete purchase and check out”. In the Setup user interface, this text is shown in the Activity Message column of Identity Verification History. The `description` must be 128 characters or fewer. If you provide a value that’s longer, it’s truncated to 128 characters.

Return Value
Type: Boolean

Usage
If the key is invalid or doesn’t exist, this method throws an invalid parameter value exception or a no data found exception, respectively. If the current user exceeds the maximum of 10 token validation attempts, this method throws a security exception.

**validateTotpTokenForUser(totpCode)**
Deprecated. Use `validateTotpTokenForUser(totpCode, description)` instead.

**Signature**

```java
public static Boolean validateTotpTokenForUser(String totpCode)
```
Parameters

totpCode
  Type: String
  The time-based one-time password (TOTP) code to validate.

Return Value
Type: Boolean

Usage
If the current user does not have a TOTP code, this method throws an exception. If the current user has attempted too many validations, this method throws an exception.

`validateTotpTokenForUser(totpCode, description)`
Indicates whether a time-based one-time password (TOTP) code (token) is valid for the current user.

Signature
`public static Boolean validateTotpTokenForUser(String totpCode, String description)`

Parameters

totpCode
  Type: String
  The time-based one-time password (TOTP) code to validate.
description
  Type: String
  The custom description that describes the activity requiring identity verification; for example, “Complete purchase and check out”. This text appears to users when they verify their identity in Salesforce and, if they use Salesforce Authenticator version 2 or later, in the Salesforce Authenticator mobile app. In addition, in the Setup user interface, this text is shown in the Activity Message column of Identity Verification History. The `description` must be 128 characters or fewer. If you provide a value that’s longer, it’s truncated to 128 characters.

Return Value
Type: Boolean

Usage
If the current user does not have a TOTP code, or if the current user has attempted too many validations, this method throws an exception.

`verifyDeviceFlow(userCode, startUrl)`
Verifies the user code entered during the device authentication flow and redirects users to the OAuth approval page. If users aren’t logged in, they must log in. After successful login, users are prompted to allow the device to access Salesforce data.
Signature

```java
public static System.PageReference verifyDeviceFlow(String userCode, String startUrl)
```

Parameters

**userCode**
- Type: String
- Human-readable user code provided to the user by Salesforce. The user must enter this code at the verification URL to approve device access to Salesforce data.

**startURL**
- Type: String
- The URL for the page that the user is redirected to after successful login and approval of the device to access Salesforce data. If you don’t specify a start URL, the user is redirected to the Home page.

Return Value

Type: `System.PageReference`

Usage

Include this method in the Apex controller when creating a custom Visualforce User Code Verification page for the OAuth 2.0 device authentication flow. This method verifies the user code, prompts the user to log in as needed, and prompts the user to allow the device access to Salesforce data. Upon successful verification and authentication, the user is redirected to the page defined by the start URL.

SessionLevel Enum

An `Auth.SessionLevel` enum value is used by the `SessionManagement.setSessionLevel` method.

Namespace

`Auth`

Enum Values

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOW</td>
<td>The user’s security level for the current session meets the lowest requirements.</td>
</tr>
<tr>
<td>STANDARD</td>
<td>The user’s security level for the current session meets the Standard requirements set in the current organization Session Security Levels.</td>
</tr>
<tr>
<td>HIGH_ASSURANCE</td>
<td>The user’s security level for the current session meets the High Assurance requirements set in the current organization Session Security Levels.</td>
</tr>
</tbody>
</table>

Note: This low level is not available, nor used, in the Salesforce UI. User sessions through the Salesforce UI are either standard or high assurance. You can set this level using the API, but users assigned this level will experience unpredictable and reduced functionality in their Salesforce organization.
Usage

With session-level security, you control user access to features that support it, such as connected apps and reporting. For example, you can customize an organization’s Session Settings to require users to log in with multi-factor authentication (MFA) to get a High Assurance session. Then, you can restrict access to a specific connected app by requiring a High Assurance session level in the settings for the connected app.

UserData Class

Stores user information for Auth.RegistrationHandler.

Namespace

Auth

IN THIS SECTION:

UserData Constructors

UserData Properties

UserData Constructors

The following are constructors for UserData.

IN THIS SECTION:

UserData(identifier, firstName, lastName, fullName, email, link, userName, locale, provider, siteLoginUrl, attributeMap)

Creates a new instance of the Auth.UserData class using the specified arguments.

Signature

public UserData(String identifier, String firstName, String lastName, String fullName, String email, String link, String userName, String locale, String provider, String siteLoginUrl, Map<String,String> attributeMap)

Parameters

identifier

Type: String

An identifier from the third party for the authenticated user, such as the Facebook user number or the Salesforce user ID.

firstName

Type: String

The first name of the authenticated user, according to the third party.
lastName
Type: String
The last name of the authenticated user, according to the third party.

fullName
Type: String
The full name of the authenticated user, according to the third party.

email
Type: String
The email address of the authenticated user, according to the third party.

link
Type: String
A stable link for the authenticated user such as https://www.facebook.com/MyUsername.

userName
Type: String
The username of the authenticated user in the third party.

locale
Type: String
The standard locale string for the authenticated user.

provider
Type: String
The service used to log in, such as Facebook or Janrain.

siteLoginUrl
Type: String
The site login page URL passed in if used with a site; null otherwise.

attributeMap
Type: Map<String, String>
A map of data from the third party, in case the handler has to access non-standard values. For example, when using Janrain as a provider, the fields Janrain returns in its accessCredentials dictionary are placed into the attributeMap. These fields vary by provider.

UserData Properties
The following are properties for UserData.

IN THIS SECTION:

identifier
An identifier from the third party for the authenticated user, such as the Facebook user number or the Salesforce user ID.

firstName
The first name of the authenticated user, according to the third party.

lastName
The last name of the authenticated user, according to the third party.
fullName
The full name of the authenticated user, according to the third party.

e-mail
The email address of the authenticated user, according to the third party.

link
A stable link for the authenticated user such as https://www.facebook.com/MyUsername.

username
The username of the authenticated user in the third party.

locale
The standard locale string for the authenticated user.

derector
The service used to log in, such as Facebook or Janrain.

siteLoginUrl
The site login page URL passed in if used with a site; null otherwise.

attributeMap
A map of data from the third party, in case the handler has to access non-standard values. For example, when using Janrain as a provider, the fields Janrain returns in its accessCredentials dictionary are placed into the attributeMap. These fields vary by provider.

**identifier**
An identifier from the third party for the authenticated user, such as the Facebook user number or the Salesforce user ID.

**Signature**
```java
public String identifier {get; set;}
```

**Property Value**
Type: String

**firstName**
The first name of the authenticated user, according to the third party.

**Signature**
```java
public String firstName {get; set;}
```

**Property Value**
Type: String

**lastName**
The last name of the authenticated user, according to the third party.
Signature

public String lastName {get; set;}

Property Value
Type: String

**fullName**
The full name of the authenticated user, according to the third party.

Signature

public String fullName {get; set;}

Property Value
Type: String

**email**
The email address of the authenticated user, according to the third party.

Signature

public String email {get; set;}

Property Value
Type: String

**link**
A stable link for the authenticated user such as https://www.facebook.com/MyUsername.

Signature

public String link {get; set;}

Property Value
Type: String

**username**
The username of the authenticated user in the third party.

Signature

public String username {get; set;}

166
Property Value
Type: String

locale
The standard locale string for the authenticated user.

Signature
public String locale {get; set;}

Property Value
Type: String

provider
The service used to log in, such as Facebook or Janrain.

Signature
public String provider {get; set;}

Property Value
Type: String

siteLoginUrl
The site login page URL passed in if used with a site; null otherwise.

Signature
public String siteLoginUrl {get; set;}

Property Value
Type: String

attributeMap
A map of data from the third party, in case the handler has to access non-standard values. For example, when using Janrain as a provider, the fields Janrain returns in its accessCredentials dictionary are placed into the attributeMap. These fields vary by provider.

Signature
public Map<String, String> attributeMap {get; set;}
Property Value
Type: Map<String, String>

VerificationMethod Enum

Contains the different ways users can identify themselves when logging in. You can use it to implement mobile-centric passwordless login pages and to self-register (and deregister) verification methods.

Usage

The enum value is an argument in `System.Site.passwordlessLogin`, `System.UserManagement.registerVerificationMethod`, and `System.UserManagement.deregisterVerificationMethod` on page 3563 methods. The value indicates the method used to verify a user’s identity.

Enum Values

The following are the values of the `Auth.VerificationMethod` enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BUILT_IN_AUTHENTICATOR</td>
<td>Identity verified with a built-in authenticator.</td>
</tr>
<tr>
<td>EMAIL</td>
<td>Identity verified with a verification code sent in an email message.</td>
</tr>
<tr>
<td>PASSWORD</td>
<td>Identity verified with a password.</td>
</tr>
<tr>
<td>SALESFORCE_AUTHENTICATOR</td>
<td>Identity verified by Salesforce Authenticator.</td>
</tr>
<tr>
<td>SECURITY_KEY</td>
<td>Identity verified by a WebAuthn-compatible physical security key. Includes all security keys registered or used after Summer ’22.</td>
</tr>
<tr>
<td>SMS</td>
<td>Identity verified with a verification code sent via SMS message.</td>
</tr>
<tr>
<td>TOTP</td>
<td>Identity verified with a time-based one-time password (TOTP).</td>
</tr>
<tr>
<td>U2F</td>
<td>Identity verified by a U2F physical security key, such as a YubiKey.</td>
</tr>
</tbody>
</table>

Note: For U2F security keys registered or used after Summer ’22, use SECURITY_KEY instead.

VerificationPolicy Enum

The `Auth.VerificationPolicy` enum contains an identity verification policy value used by the `SessionManagement.generateVerificationUrl` method.

Usage

The enum value is an argument in the `SessionManagement.generateVerificationUrl` method. The value indicates the session security policy required to initiate identity verification for the user’s session.
Enum Values

The Auth.VerificationPolicy enum has this value.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HIGH_ASSURANCE</td>
<td>The security level for the user’s current session must be High Assurance.</td>
</tr>
</tbody>
</table>

VerificationResult Class

Contains the result of a verification challenge that you invoke when you create your own Verify page. The challenge can be initiated by either the System.UserManagement.verifyPasswordlessLogin or System.UserManagement.verifySelfRegistration method.

Namespace

Auth

Usage

When users sign up for or log in to your Experience Cloud site with an email address or phone number, Salesforce sends them a verification code. At the same time, Salesforce generates the Verify page for users to enter the code to verify their identity. You can replace the Salesforce-generated Verify page with one that you create with Visualforce. Then invoke the verification challenge and, if the verification code is entered correctly, log in the user. For sign-up, you use the System.UserManagement.verifySelfRegistration method. For passwordless login, you use the System.UserManagement.verifyPasswordlessLogin method. The methods return the verification result, which contains the message displayed as a result of the challenge. This message also indicates whether the challenge is successful and where to direct the user when the verification code is entered correctly.

Example

This code contains the result of a verification challenge that registers a new user.

```java
String id = System.UserManagement.initSelfRegistration
            (Auth.VerificationMethod.SMS, user);
    Auth.VerificationResult res = System.UserManagement.verifySelfRegistration
            (Auth.VerificationMethod.SMS, id, '123456', null);
    if(res.success == true){
        //redirect
    }
```

IN THIS SECTION:

  - VerificationResult Constructor
  - VerificationResult Properties
  - VerificationResult Method

VerificationResult Constructor

VerificationResult has the following constructor.
IN THIS SECTION:

VerificationResult(redirect, success, message)

Creates an instance of the VerificationResult class that contains the verification result from System.UserManagement.verifySelfRegistration.

VerificationResult(redirect, success, message)

Creates an instance of the VerificationResult class that contains the verification result from System.UserManagement.verifySelfRegistration.

Signature

public VerificationResult(System.PageReference redirect, Boolean success, String message)

Parameters

redirect
Type: System.PageReference
Where user is directed upon successful verification.

success
Type: Boolean
Indicates whether verification succeeded.

message
Type: String
Message that displays as a result of a verification challenge.

VerificationResult Properties

The following are properties for VerificationResult.

IN THIS SECTION:

message
Message that displays as a result of a verification challenge. SUCCESS if the identity verification is successful. Other values are FAILURE, PENDING, RATE_LIMITED, or FAILURE_REPORT.

redirect
Where the user is directed after entering the verification code successfully, for example, the Experience Cloud site’s home page or location specified by the start URL.

success
The verification challenge is successful.

message
Message that displays as a result of a verification challenge. SUCCESS if the identity verification is successful. Other values are FAILURE, PENDING, RATE_LIMITED, or FAILURE_REPORT.
Signature
public String message {get; set;}

Property Value
Type: String

redirect
Where the user is directed after entering the verification code successfully, for example, the Experience Cloud site's home page or location specified by the start URL.

Signature
public System.PageReference redirect {get; set;}

Property Value
Type: System.PageReference

success
The verification challenge is successful.

Signature
public Boolean success {get; set;}

Property Value
Type: Boolean

VerificationResult Method
VerificationResult has the following method.

IN THIS SECTION:
  clone()
  Duplicates the Auth.VerificationResult object.

clone()
Duplicates the Auth.VerificationResult object.

Signature
public Object clone()
## Return Value

Type: VerificationResult

## Auth Exceptions

The `Auth` namespace contains some exception classes.

All exception classes support built-in methods for returning the error message and exception type. See Exception Class and Built-In Exceptions.

The `Auth` namespace contains the following exception.

<table>
<thead>
<tr>
<th>Exception</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auth.AuthProviderPluginException</td>
<td>Throw this exception to indicate that an error occurred when using the auth provider plug-in. Use to display a custom error message to the user. To get the error message and write it to debug log, use the <code>String</code> <code>getMessage()</code> method.</td>
</tr>
<tr>
<td>Auth.ConnectedAppPluginException</td>
<td>Throw this exception to indicate that an error occurred while running the custom behavior for a connected app. To get the error message and write it to debug log, use the <code>String</code> <code>getMessage()</code> method.</td>
</tr>
<tr>
<td>Auth.DiscoveryCustomErrorException</td>
<td>Throw this exception to customize error messages that appear on Discovery logins and Configurable Self-Registration pages. An error message can have up to 200 characters. Use custom error exceptions to localize error messages. Include this exception in:</td>
</tr>
<tr>
<td></td>
<td>• Auth.MyDomainLoginDiscoveryHandler to show a custom error message on the My Domain login page</td>
</tr>
<tr>
<td></td>
<td>• Auth.LoginDiscoveryHandler to show an error message on the Experience Cloud site login page</td>
</tr>
<tr>
<td></td>
<td>• Auth.ConfigurableSelfRegHandler to show an error message on the Experience Cloud site self-registration Verify page</td>
</tr>
<tr>
<td></td>
<td>The Verify page shows up if you configured self-registration with either an Email or Text Message verification method. If you didn’t set up sign-up with a verification method, the error message appears on the self-registration page.</td>
</tr>
<tr>
<td></td>
<td>To get the error message and write it to debug log, use the <code>String</code> <code>getMessage()</code> method.</td>
</tr>
<tr>
<td>Auth.JWTBearerTokenExchange.JWTBearerTokenExchangeException</td>
<td>Throw this exception to indicate a problem with the response from the token endpoint in the JWTBearerTokenExchange class. This exception occurs during the OAuth 2.0 JWT bearer token flow when the HTTP response:</td>
</tr>
<tr>
<td></td>
<td>• Fails to return an access token</td>
</tr>
<tr>
<td></td>
<td>• Isn’t in JSON format</td>
</tr>
<tr>
<td></td>
<td>• Returns a response code other than a 200 “OK” success code</td>
</tr>
<tr>
<td></td>
<td>To get the error message and write it to debug log, use the <code>String</code> <code>getMessage()</code> method.</td>
</tr>
<tr>
<td>Auth&gt;LoginDiscoveryException</td>
<td>Throw this exception to indicate that an error occurred when executing the Login Discovery handler. For an example, see LoginDiscoveryHandler Example Implementation. To get the error message and write it to debug log, use the <code>String</code> <code>getMessage()</code> method.</td>
</tr>
</tbody>
</table>
**Exception**  
**Description**  

<table>
<thead>
<tr>
<th>Exception</th>
<th>Description</th>
</tr>
</thead>
</table>
| Auth.VerificationException | Throw this exception to trigger verification based on the passed-in policy. You can throw this exception in an Apex trigger or Visualforce controller. The system automatically sends you to the verification endpoint, if possible.  

**Note:** You can't catch this exception. The exception immediately triggers the verification.  

---

**Examples**

This example uses Auth.ProviderPluginException to throw a custom exception in a custom authentication provider implementation. Use this exception if you want the end user to see a specific message, passing in the error message as a parameter. If you use another exception, users see a standard Salesforce error message.

```apex
global override Auth.OAuthRefreshResult refresh(Map<string,string> authProviderConfiguration, String refreshToken)
{
    HttpRequest req = new HttpRequest();
    String accessToken = null;
    String error = null;
    try {
        // DEVELOPER TODO: Make a refresh token flow using refreshToken passed in as an argument to get the new access token
        // accessToken = ...
    } catch (System.CalloutException e) {
        error = e.getMessage();
    }
    catch(Exception e) {
        error = e.getMessage();
        throw new Auth.AuthProviderPluginException('My custom error');
    }

    return new Auth.OAuthRefreshResult(accessToken,refreshToken, error);
}
```

This example uses Auth.VerificationException to trigger verification if a user attempts to create an account without a high assurance session.

```apex
trigger testTrigger on Account (before insert) {
    Map<String, String> sessionMap = auth.SessionManagement.getCurrentSession();
    if(!sessionMap.get('SessionSecurityLevel').equals('HIGH_ASSURANCE')) {
        throw new Auth.VerificationException(
            Auth.VerificationPolicy.HIGH_ASSURANCE, 'Insert Account');
    }
}
```

---

**Cache Namespace**

The Cache namespace contains methods for managing the platform cache.

The following are the classes in the Cache namespace.
IN THIS SECTION:

CacheBuilder Interface
An interface for safely retrieving and removing values from a session or org cache. Use the interface to generate a value that you want to store in the cache. The interface checks for cache misses, which means you no longer need to check for null cache values yourself.

Org Class
Use the Cache.Org class to add, retrieve, and manage values in the org cache. Unlike the session cache, the org cache is not tied to any session and is available to the organization across requests and to all users.

OrgPartition Class
Contains methods to manage cache values in the org cache of a specific partition. Unlike the session cache, the org cache is not tied to any session. It’s available to the organization across requests and to all users.

Partition Class
Base class of Cache.OrgPartition and Cache.SessionPartition. Use the subclasses to manage the cache partition for org caches and session caches.

Session Class
Use the Cache.Session class to add, retrieve, and manage values in the session cache. The session cache is active as long as the user’s Salesforce session is valid (the user is logged in, and the session is not expired).

SessionPartition Class
Contains methods to manage cache values in the session cache of a specific partition.

Cache Exceptions
The Cache namespace contains exception classes.

Visibility Enum
Use the Cache.Visibility enumeration in the Cache.Session or Cache.Org methods to indicate whether a cached value is visible only in the value’s namespace or in all namespaces.

SEE ALSO:

Apex Developer Guide: Platform Cache

CacheBuilder Interface
An interface for safely retrieving and removing values from a session or org cache. Use the interface to generate a value that you want to store in the cache. The interface checks for cache misses, which means you no longer need to check for null cache values yourself.

Namespace
Cache

SEE ALSO:

Apex Developer Guide: Safely Cache Values with the CacheBuilder Interface
CacheBuilder Methods

The following are methods for CacheBuilder.

IN THIS SECTION:

\[doLoad(var)\]
Contains the logic that builds a cached value. You don’t call this method directly. Instead, it’s called indirectly when you reference the class that implements the CacheBuilder interface.

\[doLoad(var)\]
Contains the logic that builds a cached value. You don’t call this method directly. Instead, it’s called indirectly when you reference the class that implements the CacheBuilder interface.

Signature

```
public Object doLoad(String var)
```

Parameters

\[var\]
Type: String

A case-sensitive string value used to build a cached value. This parameter is also used as part of the unique key that identifies the cached value.

Return Value

Type: Object

The value that was cached. Cast the return value to the appropriate type.

CacheBuilder Example Implementation

This example creates a class called UserInfoCache that implements the CacheBuilder interface. The class caches the results of a SOQL query run against the User object.

```
class UserInfoCache implements Cache.CacheBuilder {
    public Object doLoad(String userid) {
        User u = (User)[SELECT Id, IsActive, username FROM User WHERE id =: userid];
        return u;
    }
}
```

This example gets a cached User record based on a user ID. If the value exists in the org cache, it is returned. If the value doesn’t exist, the doLoad(String var) method is re-executed, and the new value is cached and returned.

```
User batman = (User) Cache.Org.get(UserInfoCache.class, '00541000000ek4c');
```
Org Class

Use the Cache.Org class to add, retrieve, and manage values in the org cache. Unlike the session cache, the org cache is not tied to any session and is available to the organization across requests and to all users.

Namespace

Cache

Usage

Cache Key Format

This table lists the format of the key parameter that some methods in this class take, such as put, get, and contains.

<table>
<thead>
<tr>
<th>Key Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>namespace.partition.key</td>
<td>Fully qualified key name.</td>
</tr>
<tr>
<td>key</td>
<td>Refers to a partition marked as default when the namespace.partition prefix is omitted.</td>
</tr>
<tr>
<td>local.partition.key</td>
<td>Use the local prefix to refer to the org’s namespace when the org doesn’t have a namespace defined. If the org has a namespace defined, the local prefix also refers to that org’s namespace.</td>
</tr>
</tbody>
</table>

Note:

- If no default partition is specified in the org, calling a cache method without fully qualifying the key name causes a Cache.Org.OrgCacheException to be thrown.
- The local prefix in an installed managed package refers to the namespace of the subscriber org and not the package’s namespace. The cache put calls aren’t allowed in a partition that the invoking class doesn’t own.

Example

This class is the controller for a sample Visualforce page (shown in the subsequent code sample). The cached values are initially added to the cache by the init() method, which the Visualforce page invokes when it loads through the action attribute. The cache keys don’t contain the namespace.partition prefix. They all refer to the default partition in your org. To run this sample, create a partition and mark it as default.

The Visualforce page contains four output components. These components call get methods on the controller that returns the following values from the cache: a date, data based on the MyData inner class, a counter, a text value, and a list. The size of the list is also returned.

The Visualforce page also contains two buttons. The Rerender button invokes the go() method on the controller. This method increases the values of the counter and the custom data in the cache. When you click Rerender, the two counters increase by one each time. The go() method retrieves the values of these counters from the cache, increments their values by one, and stores them again in the cache.

The Remove datetime Key button deletes the date-time value (with key datetime) from the cache. As a result, the value next to Cached datetime: is cleared on the page.
public class OrgCacheController {

    // Inner class.
    // Used as the data type of a cache value.
    class MyData {
        public String value { get; set; }
        public Integer counter { get; set; }

        public MyData(String value) {
            this.value = value;
            this.counter = 0;
        }

        public void inc() {
            counter++;
        }

        override public String toString() {
            return this.value + ':' + this.counter;
        }
    }

    // Apex List.
    // Used as the data type of a cached value.
    private List<String> numbers =
        new List<String> { 'ONE', 'TWO', 'THREE', 'FOUR', 'FIVE' };

    // Constructor of the controller for the Visualforce page.
    public OrgCacheController() {
    }

    // Adds various values to the cache.
    // This method is called when the Visualforce page loads.
    public void init() {
        // All key values are not qualified by the namespace.partition
        // prefix because they use the default partition.

        // Add counter to the cache with initial value of 0
        // or increment it if it's already there.
        if (!Cache.Org.contains('counter')) {
            Cache.Org.put('counter', 0);
        } else {
            Cache.Org.put('counter', getCounter() + 1);
        }

        // Add the datetime value to the cache only if it's not already there.
        if (!Cache.Org.contains('datetime')) {
            DateTime dt = DateTime.now();
            Cache.Org.put('datetime', dt);
        }
    }
}
// Add the custom data to the cache only if it's not already there.
if (!Cache.Org.contains('data')) {
    Cache.Org.put('data', new MyData('Some custom value'));
}

// Add a list of number to the cache if not already there.
if (!Cache.Org.contains('list')) {
    Cache.Org.put('list', numbers);
}

// Add a string value to the cache if not already there.
if (!Cache.Org.contains('output')) {
    Cache.Org.put('output', 'Cached text value');
}

// Return counter from the cache.
public Integer getCounter() {
    return (Integer)Cache.Org.get('counter');
}

// Return datetime value from the cache.
public String getCachedDatetime() {
    DateTime dt = (DateTime)Cache.Org.get('datetime');
    return dt != null ? dt.format() : null;
}

// Return cached value whose type is the inner class MyData.
public String getCachedData() {
    MyData mydata = (MyData)Cache.Org.get('data');
    return mydata != null ? mydata.toString() : null;
}

// Return output from the cache.
public String getOutput() {
    return (String)Cache.Org.get('output');
}

// Return list from the cache.
public List<String> getList() {
    return (List<String>)Cache.Org.get('list');
}

// Method invoked by the Rerender button on the Visualforce page.
// Updates the values of various cached values.
// Increases the values of counter and the MyData counter if those
// cache values are still in the cache.
public PageReference go() {
    // Increase the cached counter value or set it to 0
    // if it's not cached.
    if (Cache.Org.contains('counter')) {
        Cache.Org.put('counter', getCounter() + 1);
    } else {
        Cache.Org.put('counter', 0);
    }
}
Get the custom data value from the cache.
MyData d = (MyData)Cache.Org.get('data');
// Only if the data is already in the cache, update it.
if (Cache.Org.contains('data')) {
    d.inc();
    Cache.Org.put('data', d);
}
return null;

// Method invoked by the Remove button on the Visualforce page.
// Removes the datetime cached value from the org cache.
public PageReference remove() {
    Cache.Org.remove('datetime');
    return null;
}

This is the Visualforce page that corresponds to the OrgCacheController class.

This is the output of the page after clicking the Rerender button twice. The counter value could differ in your case if a key named counter was already in the cache before running this sample.

Cached datetime: 8/11/2015 1:58 PM
Cached data: Some custom value: 2
Cached counter: 2
Output: Cached text value
Repeat: ONE TWO THREE FOUR FIVE
List size: 5
IN THIS SECTION:

Org Constants

The Org class provides a constant that you can use when setting the time-to-live (TTL) value.

Org Methods

SEE ALSO:

*Apex Developer Guide: Platform Cache*

### Org Constants

The Org class provides a constant that you can use when setting the time-to-live (TTL) value.

<table>
<thead>
<tr>
<th>Constant</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAX_TTL_SECS</td>
<td>Represents the maximum amount of time, in seconds, to keep the cached value in the org cache.</td>
</tr>
</tbody>
</table>

### Org Methods

The following are methods for `Org`. All methods are static.

IN THIS SECTION:

- `contains(key)`
  Returns `true` if the org cache contains a cached value corresponding to the specified key.
- `contains(keys)`
  Returns `true` if the org cache contains values for the specified key entries.
- `contains(setOfKeys)`
  Returns `true` if the org cache contains values for a specified set of keys.
- `get(key)`
  Returns the cached value corresponding to the specified key from the org cache.
- `get(cacheBuilder, key)`
  Returns the cached value corresponding to the specified key from the org cache. Use this method if your cached value is a class that implements the `CacheBuilder` interface.
- `get(keys)`
  Returns the cached values corresponding to the specified set of keys from the org cache.
- `getAvgGetSize()`
  Returns the average item size of all the keys fetched from the org cache, in bytes.
- `getAvgGetTime()`
  Returns the average time taken to get a key from the org cache, in nanoseconds.
- `getAvgValueSize()`

*Deprecated and available only in API versions 49.0 and earlier.* Returns the average item size for keys in the org cache, in bytes.
getCapacity()
Returns the percentage of org cache capacity that has been used.

getKeys()
Returns a set of all keys that are stored in the org cache and visible to the invoking namespace.

getMaxGetSize()
Returns the maximum item size of all the keys fetched from the org cache, in bytes.

getMaxGetTime()
Returns the maximum time taken to get a key from the org cache, in nanoseconds.

getMaxValueSize()
Deprecated and available only in API versions 49.0 and earlier. Returns the maximum item size for keys in the org cache, in bytes.

getMissRate()
Returns the miss rate in the org cache.

getName()
Returns the name of the default cache partition.

getNumKeys()
Returns the total number of keys in the org cache.

getPartition(partitionName)
Returns a partition from the org cache that corresponds to the specified partition name.

put(key, value)
Stores the specified key/value pair as a cached entry in the org cache. The put method can write only to the cache in your org’s namespace.

put(key, value, visibility)
Stores the specified key/value pair as a cached entry in the org cache and sets the cached value’s visibility.

put(key, value, ttlSecs)
Stores the specified key/value pair as a cached entry in the org cache and sets the cached value’s lifetime.

put(key, value, ttlSecs, visibility, immutable)
Stores the specified key/value pair as a cached entry in the org cache. This method also sets the cached value’s lifetime, visibility, and whether it can be overwritten by another namespace.

remove(key)
Deletes the cached value corresponding to the specified key from the org cache.

remove(cacheBuilder, key)
Deletes the cached value corresponding to the specified key from the org cache. Use this method if your cached value is a class that implements the CacheBuilder interface.

contains(key)
Returns true if the org cache contains a cached value corresponding to the specified key.

Signature

public static Boolean contains(String key)
Parameters
key
  Type: String
  A case-sensitive string value that uniquely identifies a cached value. For information about the format of the key name, see Usage.

Return Value
Type: Boolean
true if a cache entry is found. Otherwise, false.

contains(keys)
Returns true if the org cache contains values for the specified key entries.

Signature
public static List<Boolean> contains(List<String> keys)

Parameters
keys
  Type: List<String>
  A list of keys that identifies cached values. For information about the format of the key name, see Usage.

Return Value
Type: List<Boolean>
true if the key entries are found. Otherwise, false.

contains(setOfKeys)
Returns true if the org cache contains values for a specified set of keys.

Signature
public static Map<String, Boolean> contains(Set<String> keys)

Parameters
setOfKeys
  Type: Set<String>
  A set of keys that uniquely identifies cached values. For information about the format of the key name, see Usage.

Return Value
Type: Map<String, Boolean>
Returns the cache key and corresponding Boolean value indicating that the key entry exists. The Boolean value is false if the key entry doesn't exist.
Usage
The number of input keys cannot exceed the maximum limit of 10.

Example
In this example, the code checks for the presence of multiple keys on the default partition. It fetches the cache key and the corresponding Boolean value for the key entry from the org cache of the default partition.

```apex
Set<String> keys = new Set<String>{'key1','key2','key3','key4','key5'};
Map<String,Boolean> result = Cache.Org.contains(keys);
for(String key : result.keySet()) {
    system.debug('key: ' + key);
    system.debug('Is Key Present in the cache : ' + result.get(key));
}
```

In this example, the code checks for the presence of multiple keys on different partitions. It fetches the cache key and the corresponding Boolean value for the key entry from the org cache of different partitions.

```apex
// Assuming there are three partitions p1, p2, p3 with default 'local' namespace
Set<String> keys = new Set<String>{'local.p1.key','local.p2.key','local.p3.key'};
Map<String,Boolean> result = Cache.Org.contains(keys);
for(String key : result.keySet()) {
    system.debug('key: ' + key);
    system.debug('Is Key Present in the cache : ' + result.get(key));
}
```

get (key)
Returns the cached value corresponding to the specified key from the org cache.

Signature
```
public static Object get(String key)
```

Parameters
```
key
Type: String
A case-sensitive string value that uniquely identifies a cached value. For information about the format of the key name, see Usage.
```

Return Value
Type: Object
The cached value as a generic object type. Cast the returned value to the appropriate type.

Usage
Because Cache.Org.get() returns an object, cast the returned value to a specific type to facilitate use of the returned value.

```apex
// Get a cached value
Object obj = Cache.Org.get('ns1.partition1.orderDate');
```
Cast return value to a specific data type

```java
DateTime dt2 = (DateTime)obj;
```

If a `Cache.Org.get()` call doesn’t find the referenced key, it returns `null`.

**get(cacheBuilder, key)**

Returns the cached value corresponding to the specified key from the org cache. Use this method if your cached value is a class that implements the `CacheBuilder` interface.

**Signature**

```java
public static Object get(System.Type cacheBuilder, String key)
```

**Parameters**

- **cacheBuilder**
  - Type: `System.Type`
  - The Apex class that implements the `CacheBuilder` interface.

- **key**
  - Type: `String`
  - A case-sensitive string value that, combined with the class name corresponding to the `cacheBuilder` parameter, uniquely identifies a cached value.

**Return Value**

- Type: `Object`
  - The cached value as a generic object type. Cast the returned value to the appropriate type.

**Usage**

Because `Cache.Org.get(cacheBuilder, key)` returns an object, cast the returned value to a specific type to facilitate use of the returned value.

```java
return ((DateTime)Cache.Org.get(DateCache.class, 'datetime')).format();
```

**get(keys)**

Returns the cached values corresponding to the specified set of keys from the org cache.

**Signature**

```java
public static Map<String, Object> get(Set<String> keys)
```

**Parameters**

- **keys**
  - Type: `Set<String>`
A set of keys that uniquely identify cached values. For information about the format of the key name, see Usage.

Return Value
Type: Map <String, Object>
Returns the cache key and corresponding value. Returns null when no corresponding value is found for an input key.

Usage
The number of input keys cannot exceed the maximum limit of 10.

Examples
Fetch multiple keys from the org cache of the default partition.

```java
Set<String> keys = new Set<String>{'key1','key2','key3','key4','key5'};
Map<String, Object> result = Cache.Org.get(keys);
for(String key : result.keySet()) {
    system.debug('key: ' + key);
    system.debug('value: ' + result.get(key));
}
```

Fetch multiple keys from the org cache of different partitions.

```java
// Assuming there are three partitions p1, p2, p3 with default 'local' namespace
Set<String> keys = new Set<String>{'local.p1.key','local.p2.key', 'local.p3.key'};
Map<String, Object> result = Cache.Org.get(keys);
for(String key : result.keySet()) {
    system.debug('key: ' + key);
    system.debug('value: ' + result.get(key));
}
```

getAvgGetSize()
Returns the average item size of all the keys fetched from the org cache, in bytes.

Signature
```
public static Long getAvgGetSize()
```

Return Value
Type: Long

Example
In this example the following keys and their corresponding value sizes are inserted. The code then fetches the keys: key 1, key 2, key 3 and key 4 and returns the average item size of the fetched keys.

<table>
<thead>
<tr>
<th>Key</th>
<th>Key Value Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>key 1</td>
<td>42</td>
</tr>
</tbody>
</table>
// Inserting keys key1, key2, key3, key4, key5
Cache.Org.put('key1', 'value1');
Cache.Org.put('key2', 'value2');
Cache.Org.put('key3', 'this is a big value !!!');
Cache.Org.put('key4', 4);
Cache.Org.put('key5', 5);

// Fetching keys - key1, key2, key3, key4
Object v1 = Cache.Org.get('key1');
Object v2 = Cache.Org.get('key2');
Object v3 = Cache.Org.get('key3');
Object v4 = Cache.Org.get('key4');

// Fetching average get size
Long val = Cache.Org.getAvgGetSize();
// Avg item size returned is 44 (average of 42(key1), 42(key2), 58(key3) and 36(key4)
// keys that were fetched)
System.debug('Avg Get Size :' + val);

getAvgGetTime()
Returns the average time taken to get a key from the org cache, in nanoseconds.

Signature
public static Long getAvgGetTime()

Return Value
Type: Long

getAvgValueSize()
Deprecated and available only in API versions 49.0 and earlier. Returns the average item size for keys in the org cache, in bytes.

Signature
public static Long getAvgValueSize()
Return Value
Type: Long

getCapacity()
Returns the percentage of org cache capacity that has been used.

Signature
public static Double getCapacity()

Return Value
Type: Double
Used cache as a percentage number.

getKeys()
Returns a set of all keys that are stored in the org cache and visible to the invoking namespace.

Signature
public static Set<String> getKeys()

Return Value
Type: Set<String>
A set containing all cache keys.

getMaxGetSize()
Returns the maximum item size of all the keys fetched from the org cache, in bytes.

Signature
public static Long getMaxGetSize()

Return Value
Type: Long

Example
In this example the following keys and their corresponding value sizes are inserted. The code fetches the keys: key 1, key 2 and key 4 and returns the maximum key value size from the fetched keys.

<table>
<thead>
<tr>
<th>Key</th>
<th>Key Value Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>key 1</td>
<td>42</td>
</tr>
</tbody>
</table>
### getMaxSizeGetTime

Returns the maximum time taken to get a key from the org cache, in nanoseconds.

**Signature**

```java
public static Long getMaxGetTime()
```

**Return Value**

Type: `Long`

### getMaxValueSize

Deprecated and available only in API versions 49.0 and earlier. Returns the maximum item size for keys in the org cache, in bytes.

**Signature**

```java
public static Long getMaxValueSize()
```

**Return Value**

Type: `Long`
getMissRate()  
Returns the miss rate in the org cache.

Signature  
public static Double getMissRate()  

Return Value  
Type: Double

getName()  
Returns the name of the default cache partition.

Signature  
public String getName()  

Return Value  
Type: String  
The name of the default cache partition.

getNumKeys()  
Returns the total number of keys in the org cache.

Signature  
public static Long getNumKeys()  

Return Value  
Type: Long

getPartition(partitionName)  
Returns a partition from the org cache that corresponds to the specified partition name.

Signature  
public static cache.OrgPartition getPartition(String partitionName)  

Parameters  
partitionName  
Type: String  
A partition name that is qualified by the namespace, for example, namespace.partition.
Return Value
Type: Cache.OrgPartition

Example
After you get the org partition, you can add and retrieve the partition’s cache values.

```java
// Get partition
// Retrieve cache value from the partition
if (orgPart.contains('BookTitle')) {
    String cachedTitle = (String)orgPart.get('BookTitle');
}
// Add cache value to the partition
orgPart.put('OrderDate', Date.today());
// Or use dot notation to call partition methods
String cachedAuthor = (String)Cache.Org.getPartition('myNs.myPartition').get('BookAuthor');
```

**put(key, value)**
Stores the specified key/value pair as a cached entry in the org cache. The `put` method can write only to the cache in your org’s namespace.

**Signature**
```java
public static void put(String key, Object value)
```

**Parameters**
- **key**
  - Type: String
    - A case-sensitive string value that uniquely identifies a cached value. For information about the format of the key name, see Usage.
- **value**
  - Type: Object
    - The value to store in the cache. The cached value must be serializable.

**Return Value**
Type: void

**put(key, value, visibility)**
Stores the specified key/value pair as a cached entry in the org cache and sets the cached value’s visibility.

**Signature**
```java
public static void put(String key, Object value, Cache.Visibility visibility)
```
Parameters

key
Type: String
A case-sensitive string value that uniquely identifies a cached value. For information about the format of the key name, see Usage.

value
Type: Object
The value to store in the cache. The cached value must be serializable.

visibility
Type: Cache.Visibility
Indicates whether the cached value is available only to Apex code that is executing in the same namespace or to Apex code executing from any namespace.

Return Value
Type: void

put(key, value, ttlSecs)
Stores the specified key/value pair as a cached entry in the org cache and sets the cached value’s lifetime.

Signature

public static void put(String key, Object value, Integer ttlSecs)

Parameters

key
Type: String
A case-sensitive string value that uniquely identifies a cached value. For information about the format of the key name, see Usage.

ttlSecs
Type: Integer
The amount of time, in seconds, to keep the cached value in the org cache. The maximum is 172,800 seconds (48 hours). The minimum value is 300 seconds or 5 minutes. The default value is 86,400 seconds (24 hours).

Return Value
Type: void

put(key, value, ttlSecs, visibility, immutable)
Stores the specified key/value pair as a cached entry in the org cache. This method also sets the cached value’s lifetime, visibility, and whether it can be overwritten by another namespace.
public static void put(String key, Object value, Integer ttlSecs, cache.Visibility visibility, Boolean immutable)

Parameters

key
Type: String
A case-sensitive string value that uniquely identifies a cached value. For information about the format of the key name, see Usage.

value
Type: Object
The value to store in the cache. The cached value must be serializable.

ttlSecs
Type: Integer
The amount of time, in seconds, to keep the cached value in the org cache. The maximum is 172,800 seconds (48 hours). The minimum value is 300 seconds or 5 minutes. The default value is 86,400 seconds (24 hours).

visibility
Type: Cache.Visibility
Indicates whether the cached value is available only to Apex code that is executing in the same namespace or to Apex code executing from any namespace.

immutable
Type: Boolean
Indicates whether the cached value can be overwritten by another namespace (false) or not (true).

Return Value
Type: void

remove(key)
Deletes the cached value corresponding to the specified key from the org cache.

public static Boolean remove(String key)

Parameters

key
Type: String
A case-sensitive string value that uniquely identifies a cached value. For information about the format of the key name, see Usage.

Return Value
Type: Boolean
true if the cache value was successfully removed. Otherwise, false.
**remove(cacheBuilder, key)**

Deletes the cached value corresponding to the specified key from the org cache. Use this method if your cached value is a class that implements the CacheBuilder interface.

**Signature**

```java
public static Boolean remove(System.Type cacheBuilder, String key)
```

**Parameters**

*cacheBuilder*

Type: `System.Type`

The Apex class that implements the CacheBuilder interface.

*key*

Type: `String`

A case-sensitive string value that, combined with the class name corresponding to the `cacheBuilder` parameter, uniquely identifies a cached value.

**Return Value**

Type: `Boolean`

`true` if the cache value was successfully removed. Otherwise, `false`.

**OrgPartition Class**

Contains methods to manage cache values in the org cache of a specific partition. Unlike the session cache, the org cache is not tied to any session. It’s available to the organization across requests and to all users.

**Namespace**

`Cache`

**Usage**

This class extends `Cache.Partition` and inherits all its non-static methods. Utility methods for creating and validating keys aren’t supported and can be called only from the `Cache.Partition` parent class. For a list of `Cache.Partition` methods, see `Partition Methods`.

To get an org partition, call `Cache.Org.getPartition` and pass in a fully qualified partition name, as follows.

```java
```

See `Cache Key Format for Partition Methods`.

**Example**

This class is the controller for a sample Visualforce page (shown in the subsequent code sample). The controller shows how to use the methods of `Cache.OrgPartition` to manage a cache value on a particular partition. The controller takes inputs from the Visualforce page for the partition name, key name for a counter, and initial counter value. The controller contains default values for these inputs.
When you click **Rerender** on the Visualforce page, the `go()` method is invoked and increases the counter by one. When you click **Remove Key**, the counter key is removed from the cache. The counter value gets reset to its initial value when it’s re-added to the cache.

**Note:** If another user logs in and runs this sample, the user gets the cache values that were last added or updated by the previous user. For example, if the counter value was five, the next user sees the counter value as increased to six.

```java
public class OrgPartitionController {

    // Name of a partition
    String partitionInput = 'local.myPartition';

    // Name of the key
    String counterKeyInput = 'counter';

    // Key initial value
    Integer counterInitValue = 0;

    // Org partition object
    Cache.OrgPartition orgPartition;

    // Constructor of the controller for the Visualforce page.
    public OrgPartitionController() {
    }

    // Adds counter value to the cache.
    // This method is called when the Visualforce page loads.
    public void init() {
        // Create the partition instance based on the partition name
        orgPartition = getPartition();

        // Create the partition instance based on the partition name
        // given in the Visualforce page or the default value.
        orgPartition = Cache.Org.getPartition(partitionInput);

        // Add counter to the cache with an initial value
        // or increment it if it’s already there.
        if (!orgPartition.contains(counterKeyInput)) {
            orgPartition.put(counterKeyInput, counterInitValue);
        } else {
            orgPartition.put(counterKeyInput, getCounter() + 1);
        }
    }

    // Returns the org partition based on the partition name
    // given in the Visualforce page or the default value.
    private Cache.OrgPartition getPartition() {
        if (orgPartition == null) {
            orgPartition = Cache.Org.getPartition(partitionInput);
        }

        return orgPartition;
    }

    // Return counter from the cache.
    public Integer getCounter() {
        return (Integer)getPartition().get(counterKeyInput);
    }
}
```
// Invoked by the Submit button to save input values supplied by the user.
public PageReference save() {
    // Reset the initial key value in the cache
    getPartition().put(counterKeyInput, counterInitValue);

    return null;
}

// Method invoked by the Rerender button on the Visualforce page. Updates the values of various cached values.
// Increases the values of counter and the MyData counter if those cache values are still in the cache.
public PageReference go() {
    // Get the org partition object
    orgPartition = getPartition();
    // Increase the cached counter value or set it to 0 if it's not cached.
    if (orgPartition.contains(counterKeyInput)) {
        orgPartition.put(counterKeyInput, getCounter() + 1);
    } else {
        orgPartition.put(counterKeyInput, counterInitValue);
    }

    return null;
}

// Method invoked by the Remove button on the Visualforce page. Removes the datetime cached value from the org cache.
public PageReference remove() {
    getPartition().remove(counterKeyInput);

    return null;
}

// Get and set methods for accessing variables that correspond to the input text fields on the Visualforce page.
public String getPartitionInput() {
    return partitionInput;
}

public String getCounterKeyInput() {
    return counterKeyInput;
}

public Integer getCounterInitValue() {
    return counterInitValue;
}

public void setPartitionInput(String partition) {
    this.partitionInput = partition;
}
This is the Visualforce page that corresponds to the `OrgPartitionController` class.

```apex
public void setCounterKeyInput(String keyName) {
    this.counterKeyInput = keyName;
}

public void setCounterInitValue(Integer counterValue) {
    this.counterInitValue = counterValue;
}
```

SEE ALSO:

* Apex Developer Guide: Platform Cache*

### Partition Class

Base class of `Cache.OrgPartition` and `Cache.SessionPartition`. Use the subclasses to manage the cache partition for org caches and session caches.

### Namespace

* Cache
Cache Key Format for Partition Methods

After you obtain the partition object (an instance of Cache.OrgPartition or Cache.SessionPartition), the methods to add, retrieve, and manage the cache values in a partition take the key name. The key name that you supply to these methods (get(), put(), remove(), and contains()) doesn't include the namespace.partition prefix.

IN THIS SECTION:
Partition Methods

SEE ALSO:
OrgPartition Class
SessionPartition Class
Apex Developer Guide: Platform Cache

Partition Methods

The following are methods for Partition.

IN THIS SECTION:
contains(key)  
Returns true if the cache partition contains a cached value corresponding to the specified key.
contains(setOfKeys)  
Returns true if the cache partition contains values for a specified set of keys.
createFullyQualifiedKey(namespace, partition, key)  
Generates a fully qualified key from the passed-in key components. The format of the generated key string is namespace.partition.key.
createFullyQualifiedPartition(namespace, partition)  
Generates a fully qualified partition name from the passed-in namespace and partition. The format of the generated partition string is namespace.partition.
get(key)  
Returns the cached value corresponding to the specified key from the cache partition.
get(keys)  
Returns the cached values corresponding to the specified set of keys from the cache partition.
get(cacheBuilder, key)  
Returns the cached value corresponding to the specified key from the partition cache. Use this method if your cached value is a class that implements the CacheBuilder interface.
getAvgGetSize()  
Returns the average item size of all the keys fetched from the partition, in bytes.
getAvgGetTime()  
Returns the average time taken to get a key from the partition, in nanoseconds.
getAvgValueSize()  
Deprecated and available only in API versions 49.0 and earlier. Returns the average item size for keys in the partition, in bytes.
getCapacity()  
Returns the percentage of cache used of the total capacity for this partition.

getKeys()  
Returns a set of all keys that are stored in the cache partition and visible to the invoking namespace.

getMaxGetSize()  
Returns the maximum item size of all the keys fetched from the partition, in bytes.

getMaxGetTime()  
Returns the maximum time taken to get a key from the partition, in nanoseconds.

getMaxValueSize()  
Deprecated and available only in API versions 49.0 and earlier. Returns the maximum item size for keys in the partition, in bytes.

getMissRate()  
Returns the miss rate in the partition.

getName()  
Returns the name of this cache partition.

getNumKeys()  
Returns the total number of keys in the partition.

isAvailable()  
Returns true if the Salesforce session is available. Only applies to Cache.SessionPartition. The session cache isn’t available when an active session isn’t present, such as in asynchronous Apex or code called by asynchronous Apex. For example, if batch Apex causes an Apex trigger to execute, the session cache isn’t available in the trigger because the trigger runs in asynchronous context.

put(key, value)  
Stores the specified key/value pair as a cached entry in the cache partition. The put method can write only to the cache in your org’s namespace.

put(key, value, visibility)  
Stores the specified key/value pair as a cached entry in the cache partition and sets the cached value’s visibility.

put(key, value, ttlSecs)  
Stores the specified key/value pair as a cached entry in the cache partition and sets the cached value’s lifetime.

put(key, value, ttlSecs, visibility, immutable)  
Stores the specified key/value pair as a cached entry in the cache partition. This method also sets the cached value’s lifetime, visibility, and whether it can be overwritten by another namespace.

remove(key)  
Deletes the cached value corresponding to the specified key from this cache partition.

remove(cacheBuilder, key)  
Deletes the cached value corresponding to the specified key from the partition cache. Use this method if your cached value is a class that implements the CacheBuilder interface.

validateCacheBuilder(cacheBuilder)  
Validates that the specified class implements the CacheBuilder interface.

validateKey(isDefault, key)  
Validates a cache key. This method throws a Cache.InvalidParamException if the key is not valid. A valid key is not null and contains alphanumeric characters.
validateKeyValue(isDefault, key, value)
Validates a cache key and ensures that the cache value is non-null. This method throws a Cache.InvalidParamException if the key or value is not valid. A valid key is not null and contains alphanumeric characters.

validateKeys(isDefault, keys)
Validates the specified cache keys. This method throws a Cache.InvalidParamException if the key is not valid. A valid key is not null and contains alphanumeric characters.

validatePartitionName(name)
Validates the partition name — for example, that it is not null.

contains(key)
Returns true if the cache partition contains a cached value corresponding to the specified key.

Signature
public Boolean contains(String key)

Parameters
key
Type: String
A case-sensitive string value that uniquely identifies a cached value.

Return Value
Type: Boolean
true if a cache entry is found. Otherwise, false.

contains(setOfKeys)
Returns true if the cache partition contains values for a specified set of keys.

Signature
public Map<String, Boolean> contains(Set<String> keys)

Parameters
setOfKeys
Type: Set<String>
A set of keys that uniquely identifies cached values. For information about the format of the key name, see Usage.

Return Value
Type: Map<String, Boolean>
Returns the cache key and corresponding Boolean value indicating that the key entry exists. The Boolean value is false if the key entry doesn’t exist.
Usage

The number of input keys cannot exceed the maximum limit of 10.

Example

In this example, the code checks for the presence of multiple keys on a partition. It fetches the cache key and the corresponding Boolean value for the key entry from the org cache of the partition.

```apex
// Assuming there is a partition p1 in the default 'local' namespace
Set<String> keys = new Set<String>{'key1','key2','key3','key4','key5'};
Map<String,Boolean> result = orgPart.contains(keys);
for(String key : result.keySet()) {
    system.debug('key: ' + key);
    system.debug('Is Key Present in the cache:' + result.get(key));
}
```

In this example, the code checks for the presence of multiple keys on a partition. It fetches the cache key and the corresponding Boolean value for the key entry from the session cache of the partition.

```apex
// Assuming there are three partitions p1, p2, p3 with default 'local' namespace
Set<String> keys = new Set<String>{'key1','key2','key3','key4','key5'};
Cache.SessionPartition sessionPart = Cache.Session.getPartition('local.p1');
Map<String,Boolean> result = sessionPart.contains(keys);
for(String key : result.keySet()) {
    system.debug('key: ' + key);
    system.debug('value: ' + result.get(key));
}
```

createFullyQualifiedKey(namespace, partition, key)

Generates a fully qualified key from the passed-in key components. The format of the generated key string is namespace.partition.key.

Signature

```
public static String createFullyQualifiedKey(String namespace, String partition, String key)
```

Parameters

```
namespace
    Type: String
    The namespace of the cache key.

partition
    Type: String
    The partition of the cache key.

key
    Type: String
```
The name of the cache key.

Return Value
Type: String

createFullyQualifiedPartition(namespace, partition)
Generates a fully qualified partition name from the passed-in namespace and partition. The format of the generated partition string is namespace.partition.

Signature
public static String createFullyQualifiedPartition(String namespace, String partition)

Parameters
namespace
  Type: String
  The namespace of the cache key.
partition
  Type: String
  The partition of the cache key.

Return Value
Type: String

get(key)
Returns the cached value corresponding to the specified key from the cache partition.

Signature
public Object get(String key)

Parameters
key
  Type: String
  A case-sensitive string value that uniquely identifies a cached value.

Return Value
Type: Object
The cached value as a generic object type. Cast the returned value to the appropriate type.
**get (keys)**

Returns the cached values corresponding to the specified set of keys from the cache partition.

**Signature**

```java
public Map<String, Object> get (Set<String> keys)
```

**Parameters**

- **keys**
  
  Type: Set<String>
  
  A set of keys that uniquely identify cached values. For information about the format of the key name, see **Usage**.

**Return Value**

Type: Map<String, Object>

Returns the cache key and corresponding value. Returns null when no corresponding value is found for an input key.

**Usage**

The number of input keys cannot exceed the maximum limit of 10.

**Examples**

Fetch multiple keys from the org cache of a partition.

```java
// Assuming there is a partition p1 in the default 'local' namespace

Set<String> keys = new Set<String>{'key1','key2','key3','key4','key5'};
Map<String, Object> result = orgPart.get(keys);
for(String key : result.keySet()) {
    system.debug('key: ' + key);
    system.debug('value: ' + result.get(key));
}
```

Fetch multiple keys from the session cache of a partition.

```java
// Assuming there is a partition p1 in the default 'local' namespace

Set<String> keys = new Set<String>{'key1','key2','key3','key4','key5'};
Cache.SessionPartition sessionPart = Cache.Session.getPartition('local.p1');
Map<String, Object> result = sessionPart.get(keys);
for(String key : result.keySet()) {
    system.debug('key: ' + key);
    system.debug('value: ' + result.get(key));
}
```

**get (cacheBuilder, key)**

Returns the cached value corresponding to the specified key from the partition cache. Use this method if your cached value is a class that implements the `CacheBuilder` interface.
**Signature**

```java
public Object get(System.Type cacheBuilder, String key)
```

**Parameters**

- `cacheBuilder`
  
  Type: `System.Type`
  
  The Apex class that implements the `CacheBuilder` interface.

- `key`
  
  Type: `String`
  
  A case-sensitive string value that, combined with the class name corresponding to the `cacheBuilder` parameter, uniquely identifies a cached value.

**Return Value**

Type: `Object`

The cached value as a generic object type. Cast the returned value to the appropriate type.

---

**getAvgGetSize()**

Returns the average item size of all the keys fetched from the partition, in bytes.

**Signature**

```java
public Long getAvgGetSize()
```

**Return Value**

Type: `Long`

---

**getAvgGetTime()**

Returns the average time taken to get a key from the partition, in nanoseconds.

**Signature**

```java
public Long getAvgGetTime()
```

**Return Value**

Type: `Long`

---

**getAvgValueSize()**

Deprecated and available only in API versions 49.0 and earlier. Returns the average item size for keys in the partition, in bytes.

**Signature**

```java
public Long getAvgValueSize()
```
Return Value
Type: Long

**getCapacity()**
Returns the percentage of cache used of the total capacity for this partition.

**Signature**
```
public Double getCapacity()
```

Return Value
Type: Double
Used partition cache as a percentage number.

**getKeys()**
Returns a set of all keys that are stored in the cache partition and visible to the invoking namespace.

**Signature**
```
public Set<String> getKeys()
```

Return Value
Type: Set<String>
A set containing all cache keys.

**getMaxGetSize()**
Returns the maximum item size of all the keys fetched from the partition, in bytes.

**Signature**
```
public Long getMaxGetSize()
```

Return Value
Type: Long

**getMaxGetTime()**
Returns the maximum time taken to get a key from the partition, in nanoseconds.

**Signature**
```
public Long getMaxGetTime()
```
Return Value
Type: Long

**getMaxValueSize()**

*Deprecated and available only in API versions 49.0 and earlier.* Returns the maximum item size for keys in the partition, in bytes.

**Signature**

```java
public Long getMaxValueSize()
```

Return Value
Type: Long

**getMissRate()**

Returns the miss rate in the partition.

**Signature**

```java
public Double getMissRate()
```

Return Value
Type: Double

**getName()**

Returns the name of this cache partition.

**Signature**

```java
public String getName()
```

Return Value
Type: String

The name of this cache partition.

**getNumKeys()**

Returns the total number of keys in the partition.

**Signature**

```java
public Long getNumKeys()
```
Return Value
Type: Long

**isAvailable()**
Returns `true` if the Salesforce session is available. Only applies to `Cache.SessionPartition`. The session cache isn’t available when an active session isn’t present, such as in asynchronous Apex or code called by asynchronous Apex. For example, if batch Apex causes an Apex trigger to execute, the session cache isn’t available in the trigger because the trigger runs in asynchronous context.

**Signature**

```java
public Boolean isAvailable()
```

**Return Value**
Type: Boolean

**put(key, value)**
Stores the specified key/value pair as a cached entry in the cache partition. The `put` method can write only to the cache in your org’s namespace.

**Signature**

```java
public void put(String key, Object value)
```

**Parameters**

- **key**
  Type: `String`
  A case-sensitive string value that uniquely identifies a cached value.

- **value**
  Type: `Object`
  The value to store in the cache. The cached value must be serializable.

**Return Value**
Type: void

**put(key, value, visibility)**
Stores the specified key/value pair as a cached entry in the cache partition and sets the cached value’s visibility.

**Signature**

```java
public void put(String key, Object value, cache.Visibility visibility)
```
Parameters

key
Type: String
A case-sensitive string value that uniquely identifies a cached value.

value
Type: Object
The value to store in the cache. The cached value must be serializable.

visibility
Type: Cache.Visibility
Indicates whether the cached value is available only to Apex code that is executing in the same namespace or to Apex code executing from any namespace.

Return Value
Type: void

put(key, value, ttlSecs)
Stores the specified key/value pair as a cached entry in the cache partition and sets the cached value’s lifetime.

Signature
public void put(String key, Object value, Integer ttlSecs)

Parameters

key
Type: String
A case-sensitive string value that uniquely identifies a cached value.

value
Type: Object
The value to store in the cache. The cached value must be serializable.

ttlSecs
Type: Integer
The amount of time, in seconds, to keep the cached value in the cache.

Return Value
Type: void

put(key, value, ttlSecs, visibility, immutable)
Stores the specified key/value pair as a cached entry in the cache partition. This method also sets the cached value’s lifetime, visibility, and whether it can be overwritten by another namespace.
Signature

public void put(String key, Object value, Integer ttlSecs, cache.Visibility visibility, Boolean immutable)

Parameters

key
Type: String
A case-sensitive string value that uniquely identifies a cached value.

value
Type: Object
The value to store in the cache. The cached value must be serializable.

ttlSecs
Type: Integer
The amount of time, in seconds, to keep the cached value in the cache.

visibility
Type: Cache.Visibility
Indicates whether the cached value is available only to Apex code that is executing in the same namespace or to Apex code executing from any namespace.

immutable
Type: Boolean
Indicates whether the cached value can be overwritten by another namespace (false) or not (true).

Return Value
Type: void

remove(key)
Deletes the cached value corresponding to the specified key from this cache partition.

Signature

public Boolean remove(String key)

Parameters

key
Type: String
A case-sensitive string value that uniquely identifies a cached value.

Return Value
Type: Boolean
true if the cache value was successfully removed. Otherwise, false.
**remove(cacheBuilder, key)**

Deletes the cached value corresponding to the specified key from the partition cache. Use this method if your cached value is a class that implements the CacheBuilder interface.

**Signature**

```java
public Boolean remove(System.Type cacheBuilder, String key)
```

**Parameters**

- `cacheBuilder`
  - Type: `System.Type`
  - The Apex class that implements the CacheBuilder interface.

- `key`
  - Type: `String`
  - A case-sensitive string value that, combined with the class name corresponding to the `cacheBuilder` parameter, uniquely identifies a cached value.

**Return Value**

Type: `Boolean`

- `true` if the cache value was successfully removed. Otherwise, `false`.

**validateCacheBuilder(cacheBuilder)**

Validates that the specified class implements the CacheBuilder interface.

**Signature**

```java
public static void validateCacheBuilder(System.Type cacheBuilder)
```

**Parameters**

- `cacheBuilder`
  - Type: `System.Type`
  - The class to validate.

**Return Value**

Type: void

**validateKey(isDefault, key)**

Validates a cache key. This method throws a `Cache.InvalidParamException` if the key is not valid. A valid key is not `null` and contains alphanumeric characters.
**validateKey** (isDefault, key)

**Signature**

```java
public static void validateKey(Boolean isDefault, String key)
```

**Parameters**

- **isDefault**
  - Type: `Boolean`
  - Set to `true` if the key references a default partition. Otherwise, set to `false`.

- **key**
  - Type: `String`
  - The key to validate.

**Return Value**

Type: `void`

**validateKeyValue** (isDefault, key, value)

Validates a cache key and ensures that the cache value is non-null. This method throws a `Cache.InvalidParamException` if the key or value is not valid. A valid key is not `null` and contains alphanumeric characters.

**Signature**

```java
public static void validateKeyValue(Boolean isDefault, String key, Object value)
```

**Parameters**

- **isDefault**
  - Type: `Boolean`
  - Set to `true` if the key references a default partition. Otherwise, set to `false`.

- **key**
  - Type: `String`
  - The key to validate.

- **value**
  - Type: `Object`
  - The cache value to validate.

**Return Value**

Type: `void`

**validateKeys** (isDefault, keys)

Validates the specified cache keys. This method throws a `Cache.InvalidParamException` if the key is not valid. A valid key is not `null` and contains alphanumeric characters.

**Signature**

```java
public static void validateKeys(Boolean isDefault, Keys keys)
```
**Signature**

```java
public static void validateKeys(Boolean isDefault, Set<String> keys)
```

**Parameters**

- `isDefault`
  - Type: `Boolean`
  - Set to `true` if the key references a default partition. Otherwise, set to `false`.

- `keys`
  - Type: `Set<String>`
  - A set of key string values to validate.

**Return Value**

Type: `void`

---

**validatePartitionName(name)**

Validates the partition name — for example, that it is not null.

**Signature**

```java
public static void validatePartitionName(String name)
```

**Parameters**

- `name`
  - Type: `String`
  - The name of the partition to validate.

**Return Value**

Type: `void`

---

**Session Class**

Use the `Cache.Session` class to add, retrieve, and manage values in the session cache. The session cache is active as long as the user’s Salesforce session is valid (the user is logged in, and the session is not expired).

**Namespace**

Cache

**Usage**

**Cache Key Format**

This table lists the format of the key parameter that some methods in this class take, such as `put`, `get`, and `contains`. 
<table>
<thead>
<tr>
<th>Key Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>namespace.partition.key</td>
<td>Fully qualified key name.</td>
</tr>
<tr>
<td>key</td>
<td>Refers to a partition marked as default when the namespace.partition prefix is omitted.</td>
</tr>
<tr>
<td>local.partition.key</td>
<td>Use the local prefix to refer to the org’s namespace when the org doesn’t have a namespace defined. If the org has a namespace defined, the local prefix also refers to that org’s namespace.</td>
</tr>
</tbody>
</table>

**Note:**
- If no default partition is specified in the org, calling a cache method without fully qualifying the key name causes a Cache.Session.SessionCacheException to be thrown.
- The local prefix in an installed managed package refers to the namespace of the subscriber org and not the package’s namespace. The cache put calls are not allowed in a partition that the invoking class doesn’t own.

**Example**

This class is the controller for a sample Visualforce page (shown in the subsequent code sample). The cached values are initially added to the cache by the `init()` method, which the Visualforce page invokes when it loads through the action attribute. The cache keys don’t contain the `namespace.partition` prefix. They all refer to a default partition in your org. The Visualforce page expects a partition named `myPartition`. To run this sample, create a default partition in your org with the name `myPartition`.

The Visualforce page contains four output components. The first three components call `get` methods on the controller that return the following values from the cache: a date, data based on the `MyData` inner class, and a counter. The next output component uses the `$Cache.Session` global variable to get the cached string value for the key named `output`. Next, the `$Cache.Session` global variable is used again in the Visualforce page to iterate over the elements of a cached value of type `List`. The size of the list is also returned.

The Visualforce page also contains two buttons. The Rerender button invokes the `go()` method on the controller. This method increases the values of the counter and the custom data in the cache. If you click Rerender, the two counters increase by one each time. The `go()` method retrieves the values of these counters from the cache, increments their values by one, and stores them again in the cache.

The Remove button deletes the date-time value (with key `datetime`) from the cache. As a result, the value next to `Cached datetime:` is cleared on the page.

```java
public class SessionCacheController {

    // Inner class.
    // Used as the data type of a cache value.
    class MyData {
        public String value { get; set; }
        public Integer counter { get; set; }

        public MyData(String value) {
            this.value = value;
            this.counter = 0;
        }
    }
}
```
public void inc() {
    counter++;
}

override public String toString() {
    return this.value + ': ' + this.counter;
}

// Apex List.
// Used as the data type of a cached value.
private List<String> numbers =
    new List<String> { 'ONE', 'TWO', 'THREE', 'FOUR', 'FIVE' };

// Constructor of the controller for the Visualforce page.
public SessionCacheController() {
}

// Adds various values to the cache.
// This method is called when the Visualforce page loads.
public void init() {
    // All key values are not qualified by the namespace.partition
    // prefix because they use the default partition.
    // Add counter to the cache with initial value of 0
    // or increment it if it's already there.
    if (!Cache.Session.contains('counter')) {
        Cache.Session.put('counter', 0);
    } else {
        Cache.Session.put('counter', getCounter() + 1);
    }

    // Add the datetime value to the cache only if it's not already there.
    if (!Cache.Session.contains('datetime')) {
        DateTime dt = DateTime.now();
        Cache.Session.put('datetime', dt);
    }

    // Add the custom data to the cache only if it's not already there.
    if (!Cache.Session.contains('data')) {
        Cache.Session.put('data', new MyData('Some custom value'));
    }

    // Add a list of number to the cache if not already there.
    if (!Cache.Session.contains('list')) {
        Cache.Session.put('list', numbers);
    }

    // Add a string value to the cache if not already there.
    if (!Cache.Session.contains('output')) {
        Cache.Session.put('output', 'Cached text value');
    }
}
// Return counter from the cache.
public Integer getCounter() {
    return (Integer)Cache.Session.get('counter');
}

// Return datetime value from the cache.
public String getCachedDatetime() {
    DateTime dt = (DateTime)Cache.Session.get('datetime');
    return dt != null ? dt.format() : null;
}

// Return cached value whose type is the inner class MyData.
public String getCachedData() {
    MyData mydata = (MyData)Cache.Session.get('data');
    return mydata != null ? mydata.toString() : null;
}

// Method invoked by the Rerender button on the Visualforce page.
// Updates the values of various cached values.
// Increases the values of counter and the MyData counter if those
// cache values are still in the cache.
public PageReference go() {
    // Increase the cached counter value or set it to 0
    // if it's not cached.
    if (Cache.Session.contains('counter')) {
        Cache.Session.put('counter', getCounter() + 1);
    } else {
        Cache.Session.put('counter', 0);
    }

    // Get the custom data value from the cache.
    MyData d = (MyData)Cache.Session.get('data');
    // Only if the data is already in the cache, update it.
    if (Cache.Session.contains('data')) {
        d.inc();
        Cache.Session.put('data', d);
    }

    return null;
}

// Method invoked by the Remove button on the Visualforce page.
// Removes the datetime cached value from the session cache.
public PageReference remove() {
    Cache.Session.remove('datetime');
    return null;
}

This is the Visualforce page that corresponds to the SessionCacheController class.
This is the output of the page after clicking the Rerender button twice. The counter value could differ in your case if a key named counter was already in the cache before running this sample.

Cached datetime: 8/11/2015 1:58 PM
Cached data: Some custom value: 2
Cached counter: 2
Output: Cached text value
Repeat: ONE TWO THREE FOUR FIVE
List size: 5

IN THIS SECTION:

Session Constants
The Session class provides a constant that you can use when setting the time-to-live (TTL) value.

Session Methods

SEE ALSO:
Apex Developer Guide: Platform Cache

**Session Constants**
The Session class provides a constant that you can use when setting the time-to-live (TTL) value.

<table>
<thead>
<tr>
<th>Constant</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAX_TTL_SECS</td>
<td>Represents the maximum amount of time, in seconds, to keep the cached value in the session cache.</td>
</tr>
</tbody>
</table>
Session Methods

The following are methods for Session. All methods are static.

IN THIS SECTION:

contains(key)
Returns true if the session cache contains a cached value corresponding to the specified key.

contains(setOfKeys)
Returns true if the cache contains values for a specified set of keys.

get(key)
Returns the cached value corresponding to the specified key from the session cache.

get(keys)
Returns the cached values corresponding to the specified set of keys from the session cache.

get(cacheBuilder, key)
Returns the cached value corresponding to the specified key from the session cache. Use this method if your cached value is a class that implements the CacheBuilder interface.

getAvgGetSize()
Returns the average item size of all the keys fetched from the session cache, in bytes.

getAvgGetTime()
Returns the average time taken to get a key from the session cache, in nanoseconds.

getAvgValueSize()
Deprecated and available only in API versions 49.0 and earlier. Returns the average item size for keys in the session cache, in bytes.

getCapacity()
Returns the percentage of session cache capacity that has been used.

getKeys()
Returns all keys that are stored in the session cache and visible to the invoking namespace.

getMaxGetSize()
Returns the maximum item size of all the keys fetched from the session cache, in bytes.

getMaxGetTime()
Returns the maximum time taken to get a key from the session cache, in nanoseconds.

getMaxValueSize()
Deprecated and available only in API versions 49.0 and earlier. Returns the maximum item size for keys in the session cache, in bytes.

getMissRate()
Returns the miss rate in the session cache.

getName()
Returns the name of the default cache partition.

getNumKeys()
Returns the total number of keys in the session cache.
getPartition(partitionName)
Returns a partition from the session cache that corresponds to the specified partition name.

isAvailable()
Returns true if the session cache is available for use. The session cache isn’t available when an active session isn’t present, such as in asynchronous Apex or code called by asynchronous Apex. For example, if batch Apex causes an Apex trigger to execute, the session cache isn’t available in the trigger because the trigger runs in asynchronous context.

put(key, value)
Stores the specified key/value pair as a cached entry in the session cache. The put method can write only to the cache in your org’s namespace.

put(key, value, visibility)
Stores the specified key/value pair as a cached entry in the session cache and sets the cached value’s visibility.

put(key, value, ttlSecs)
Stores the specified key/value pair as a cached entry in the session cache and sets the cached value’s lifetime.

put(key, value, ttlSecs, visibility, immutable)
Stores the specified key/value pair as a cached entry in the session cache. This method also sets the cached value’s lifetime, visibility, and whether it can be overwritten by another namespace.

remove(key)
Deletes the cached value corresponding to the specified key from the session cache.

remove(cacheBuilder, key)
Deletes the cached value corresponding to the specified key from the session cache. Use this method if your cached value is a class that implements the CacheBuilder interface.

contains(key)
Returns true if the session cache contains a cached value corresponding to the specified key.

Signature
public static Boolean contains(String key)

Parameters
key
Type: String
A case-sensitive string value that uniquely identifies a cached value. For information about the format of the key name, see Usage.

Return Value
Type: Boolean
true if a cache entry is found. Otherwise, false.

contains(setOfKeys)
Returns true if the cache contains values for a specified set of keys.
Signature

public static Map<String, Boolean> contains (Set<String> keys)

Parameters

setOfKeys
Type: Set<String>
A set of keys that uniquely identifies cached values. For information about the format of the key name, see Usage.

Return Value

Type: Map<String, Boolean>
Returns the cache key and corresponding Boolean value indicating that the key entry exists. The Boolean value is false if the key entry doesn’t exist.

Usage

The number of input keys cannot exceed the maximum limit of 10.

Example

In this example, the code checks for the presence of multiple keys on the default partition. It fetches the cache key and the corresponding Boolean value for the key entry from the session cache of the default partition.

```java
Set<String> keys = new Set<String>{'key1','key2','key3','key4','key5'};
Map<String,Boolean> result = Cache.Session.contains(keys);
for(String key : result.keySet()) {
    system.debug('key: ' + key);
    system.debug('Is Key Present in the cache : ' + result.get(key));
}
```

In this example, the code checks for the presence of multiple keys on different partitions. It fetches the cache key and the corresponding Boolean value for the key entry from the session cache of different partitions.

```java
// Assuming there are three partitions p1, p2, p3 with default 'local' namespace
Set<String> keys = new Set<String>{'local.p1.key','local.p2.key', 'local.p3.key'};
Map<String,Boolean> result = Cache.Session.contains(keys);
for(String key : result.keySet()) {
    system.debug('key: ' + key);
    system.debug('Is Key Present in the cache : ' + result.get(key));
}
```

get(key)

Returns the cached value corresponding to the specified key from the session cache.

Signature

public static Object get(String key)
Parameters

key
Type: String
A case-sensitive string value that uniquely identifies a cached value. For information about the format of the key name, see Usage.

Return Value

Type: Object
The cached value as a generic object type. Cast the returned value to the appropriate type.

Usage

Because `Cache.Session.get()` returns an object, we recommend that you cast the returned value to a specific type to facilitate use of the returned value.

```java
// Get a cached value
Object obj = Cache.Session.get('ns1.partition1.orderDate');
// Cast return value to a specific data type
DateTime dt2 = (DateTime)obj;
```

If a `Cache.Session.get()` call doesn’t find the referenced key, it returns `null`.

`get(keys)`

Returns the cached values corresponding to the specified set of keys from the session cache.

Signature

```java
public static Map<String, Object> get (Set<String> keys)
```

Parameters

keys
Type: Set<String>
A set of keys that uniquely identify cached values. For information about the format of the key name, see Usage.

Return Value

Type: Map<String, Object>
Returns the cache key and corresponding value. Returns null when no corresponding value is found for an input key.

Usage

The number of input keys cannot exceed the maximum limit of 10.
Example

Fetch multiple keys from the session cache of the default partition.

```java
Set<String> keys = new Set<String>{'key1','key2','key3','key4','key5'};
Map<String,Object> result = Cache.Session.get(keys);
for(String key : result.keySet()) {
    system.debug('key: ' + key);
    system.debug('value: ' + result.get(key));
}
```

Fetch multiple keys from the session cache of different partitions.

```java
// Assuming there are three partitions p1, p2, p3 with default 'local' namespace
Set<String> keys = new Set<String>{'local.p1.key','local.p2.key', 'local.p3.key'};
Map<String,Object> result = Cache.Session.get(keys);
for(String key : result.keySet()) {
    system.debug('key: ' + key);
    system.debug('value: ' + result.get(key));
}
```

get(cacheBuilder, key)

Returns the cached value corresponding to the specified key from the session cache. Use this method if your cached value is a class that implements the CacheBuilderInterface.

Signature

```java
public static Object get(System.Type cacheBuilder, String key)
```

Parameters

cacheBuilder

Type: System.Type

The Apex class that implements the CacheBuilderInterface.

key

Type: String

A case-sensitive string value that, combined with the class name corresponding to the cacheBuilder parameter, uniquely identifies a cached value.

Return Value

Type: Object

The cached value as a generic object type. Cast the returned value to the appropriate type.

Usage

Because Cache.Session.get(cacheBuilder, key) returns an object, cast the returned value to a specific type to facilitate use of the returned value.

```java
return ((DateTime)Cache.Session.get(DateCache.class, 'datetime')).format();
```
getAvgGetSize()  
Returns the average item size of all the keys fetched from the session cache, in bytes.

Signature

\texttt{public static Long getAvgGetSize()}

Return Value

Type: Long

gAvgGetTime()  
Returns the average time taken to get a key from the session cache, in nanoseconds.

Signature

\texttt{public static Long getAvgGetTime()}

Return Value

Type: Long

gAvgValueSize()  
Deprecated and available only in API versions 49.0 and earlier. Returns the average item size for keys in the session cache, in bytes.

Signature

\texttt{public static Long getAvgValueSize()}

Return Value

Type: Long

gCapacity()  
Returns the percentage of session cache capacity that has been used.

Signature

\texttt{public static Double getCapacity()}

Return Value

Type: Double

Used cache as a percentage number.
getKeys()
Returns all keys that are stored in the session cache and visible to the invoking namespace.

Signature
public static Set<String> getKeys()

Return Value
Type: Set<String>
A set containing all cache keys.

getMaxGetSize()
Returns the maximum item size of all the keys fetched from the session cache, in bytes.

Signature
public static Long getMaxGetSize()

Return Value
Type: Long

getMaxGetTime()
Returns the maximum time taken to get a key from the session cache, in nanoseconds.

Signature
public static Long getMaxGetTime()

Return Value
Type: Long

getMaxValueSize()
Deprecated and available only in API versions 49.0 and earlier. Returns the maximum item size for keys in the session cache, in bytes.

Signature
public static Long getMaxValueSize()

Return Value
Type: Long
**getMissRate()**

Returns the miss rate in the session cache.

**Signature**

```java
public static Double getMissRate()
```

**Return Value**

Type: Double

**getName()**

Returns the name of the default cache partition.

**Signature**

```java
public String getName()
```

**Return Value**

Type: String

The name of the default cache partition.

**getNumKeys()**

Returns the total number of keys in the session cache.

**Signature**

```java
public static Long getNumKeys()
```

**Return Value**

Type: Long

**getPartition(partitionName)**

Returns a partition from the session cache that corresponds to the specified partition name.

**Signature**

```java
public static cache.SessionPartition getPartition(String partitionName)
```

**Parameters**

- **partitionName**
  
  Type: String
  
  A partition name that is qualified by the namespace, for example, namespace.partition.
Return Value
Type: Cache.SessionPartition

Example
After you get the session partition, you can add and retrieve the partition’s cache values.

```java
// Get partition
Cache.SessionPartition sessionPart = Cache.Session.getPartition('myNs.myPartition');
// Retrieve cache value from the partition
if (sessionPart.contains('BookTitle')) {
    String cachedTitle = (String)sessionPart.get('BookTitle');
}

// Add cache value to the partition
sessionPart.put('OrderDate', Date.today());

// Or use dot notation to call partition methods
String cachedAuthor =
    (String)Cache.Session.getPartition('myNs.myPartition').get('BookAuthor');
```

**isAvailable()**
Returns true if the session cache is available for use. The session cache isn’t available when an active session isn’t present, such as in asynchronous Apex or code called by asynchronous Apex. For example, if batch Apex causes an Apex trigger to execute, the session cache isn’t available in the trigger because the trigger runs in asynchronous context.

**Signature**
```
public static Boolean isAvailable()
```

**Return Value**
Type: Boolean
true if the session cache is available. Otherwise, false.

**put(key, value)**
Stores the specified key/value pair as a cached entry in the session cache. The put method can write only to the cache in your org’s namespace.

**Signature**
```
public static void put(String key, Object value)
```

**Parameters**

- `key`
  Type: String
  A string that uniquely identifies the value to be cached. For information about the format of the key name, see Usage.
value
  Type: Object
  The value to store in the cache. The cached value must be serializable.

Return Value
Type: void

put(key, value, visibility)
Stores the specified key/value pair as a cached entry in the session cache and sets the cached value's visibility.

Signature
public static void put(String key, Object value, Cache.Visibility visibility)

Parameters
key
  Type: String
  A string that uniquely identifies the value to be cached. For information about the format of the key name, see Usage.

value
  Type: Object
  The value to store in the cache. The cached value must be serializable.

visibility
  Type: Cache.Visibility
  Indicates whether the cached value is available only to Apex code that is executing in the same namespace or to Apex code executing from any namespace.

Return Value
Type: void

put(key, value, ttlSecs)
Stores the specified key/value pair as a cached entry in the session cache and sets the cached value's lifetime.

Signature
public static void put(String key, Object value, Integer ttlSecs)

Parameters
key
  Type: String
  A string that uniquely identifies the value to be cached. For information about the format of the key name, see Usage.
value
Type: Object
The value to store in the cache. The cached value must be serializable.

ttlSecs
Type: Integer
The amount of time, in seconds, to keep the cached value in the session cache. The cached values remain in the cache as long as the Salesforce session hasn’t expired. The maximum value is 28,800 seconds or eight hours. The minimum value is 300 seconds or five minutes.

Return Value
Type: void

put(key, value, ttlSecs, visibility, immutable)
Stores the specified key/value pair as a cached entry in the session cache. This method also sets the cached value’s lifetime, visibility, and whether it can be overwritten by another namespace.

Signature
public static void put(String key, Object value, Integer ttlSecs, cache.Visibility visibility, Boolean immutable)

Parameters
key
Type: String
A string that uniquely identifies the value to be cached. For information about the format of the key name, see Usage.

value
Type: Object
The value to store in the cache. The cached value must be serializable.

ttlSecs
Type: Integer
The amount of time, in seconds, to keep the cached value in the session cache. The cached values remain in the cache as long as the Salesforce session hasn’t expired. The maximum value is 28,800 seconds or eight hours. The minimum value is 300 seconds or five minutes.

visibility
Type: Cache.Visibility
Indicates whether the cached value is available only to Apex code that is executing in the same namespace or to Apex code executing from any namespace.

immutable
Type: Boolean
Indicates whether the cached value can be overwritten by another namespace (false) or not (true).
Return Value
Type: void

**remove(key)**
Deletes the cached value corresponding to the specified key from the session cache.

**Signature**
```java
public static Boolean remove(String key)
```

**Parameters**
- `key`  
  Type: `String`  
  A case-sensitive string value that uniquely identifies a cached value. For information about the format of the key name, see Usage.

**Return Value**
Type: `Boolean`  
`true` if the cache value was successfully removed. Otherwise, `false`.

**remove(cacheBuilder, key)**
Deletes the cached value corresponding to the specified key from the session cache. Use this method if your cached value is a class that implements the CacheBuilder interface.

**Signature**
```java
public static Boolean remove(System.Type cacheBuilder, String key)
```

**Parameters**
- `cacheBuilder`  
  Type: `System.Type`  
  The Apex class that implements the CacheBuilder interface.
- `key`  
  Type: `String`  
  A case-sensitive string value that, combined with the class name corresponding to the `cacheBuilder` parameter, uniquely identifies a cached value.

**Return Value**
Type: `Boolean`  
`true` if the cache value was successfully removed. Otherwise, `false`.
SessionPartition Class
Contains methods to manage cache values in the session cache of a specific partition.

Namespace
Cache

Usage
This class extends Cache.Partition and inherits all of its non-static methods. Utility methods for creating and validating keys are not supported and can be called only from the Cache.Partition parent class. For a list of Cache.Partition methods, see Partition Methods.

To get a session partition, call Cache.Session.getPartition and pass in a fully qualified partition name, as follows.

```java
Cache.SessionPartition sessionPartition = Cache.Session.getPartition('namespace.myPartition');
```

See Cache Key Format for Partition Methods.

Example
This class is the controller for a sample Visualforce page (shown in the subsequent code sample). The controller shows how to use the methods of Cache.SessionPartition to manage a cache value on a particular partition. The controller takes inputs from the Visualforce page for the partition name, key name for a counter, and initial counter value. The controller contains default values for these inputs. When you click Rerender on the Visualforce page, the go() method is invoked and increases the counter by one. When you click Remove Key, the counter key is removed from the cache. The counter value gets reset to its initial value when it’s re-added to the cache.

```java
public class SessionPartitionController {    
  // Name of a partition in the local namespace
  String partitionInput = 'local.myPartition';
  // Name of the key
  String counterKeyInput = 'counter';
  // Key initial value
  Integer counterInitValue = 0;
  // Session partition object
  Cache.SessionPartition sessionPartition;

  // Constructor of the controller for the Visualforce page.
  public SessionPartitionController() {
  }

  // Adds counter value to the cache.
  // This method is called when the Visualforce page loads.
  public void init() {
    // Create the partition instance based on the partition name
    sessionPartition = getPartition();

    // Add counter to the cache with an initial value
    // or increment it if it's already there.
  }
}
```
if (!sessionPartition.contains(counterKeyInput)) {
    sessionPartition.put(counterKeyInput, counterInitValue);
} else {
    sessionPartition.put(counterKeyInput, getCounter() + 1);
}

// Returns the session partition based on the partition name
// given in the Visualforce page or the default value.
private Cache.SessionPartition getPartition() {
    if (sessionPartition == null) {
        sessionPartition = Cache.Session.getPartition(partitionInput);
    }

    return sessionPartition;
}

// Return counter from the cache.
public Integer getCounter() {
    return (Integer)getPartition().get(counterKeyInput);
}

// Invoked by the Submit button to save input values
// supplied by the user.
public PageReference save() {
    // Reset the initial key value in the cache
    getPartition().put(counterKeyInput, counterInitValue);

    return null;
}

// Method invoked by the Rerender button on the Visualforce page.
// Updates the values of various cached values.
// Increases the values of counter and the MyData counter if those
// cache values are still in the cache.
public PageReference go() {
    // Get the partition object
    sessionPartition = getPartition();
    // Increase the cached counter value or set it to 0
    // if it's not cached.
    if (sessionPartition.contains(counterKeyInput)) {
        sessionPartition.put(counterKeyInput, getCounter() + 1);
    } else {
        sessionPartition.put(counterKeyInput, counterInitValue);
    }

    return null;
}

// Method invoked by the Remove button on the Visualforce page.
// Removes the datetime cached value from the session cache.
public PageReference remove() {
    getPartition().remove(counterKeyInput);
}
// Get and set methods for accessing variables
// that correspond to the input text fields on
// the Visualforce page.
public String getPartitionInput() {
    return partitionInput;
}

public String getCounterKeyInput() {
    return counterKeyInput;
}

public Integer getCounterInitValue() {
    return counterInitValue;
}

public void setPartitionInput(String partition) {
    this.partitionInput = partition;
}

public void setCounterKeyInput(String keyName) {
    this.counterKeyInput = keyName;
}

public void setCounterInitValue(Integer counterValue) {
    this.counterInitValue = counterValue;
}

This is the Visualforce page that corresponds to the SessionPartitionController class.

```apex
<apex:page controller="SessionPartitionController" action="{!init}">
    <apex:form>
        <br/>Partition with Namespace Prefix: <apex:inputText value="{!partitionInput}"/>
        <br/>Counter Key Name: <apex:inputText value="{!counterKeyInput}"/>
        <br/>Counter Initial Value: <apex:inputText value="{!counterInitValue}"/>
        <apex:commandButton action="{!save}" value="Save Key Input Values"/>
    </apex:form>

    <apex:outputPanel id="output">
        <br/>Cached Counter: <apex:outputText value="{!counter}"/>
    </apex:outputPanel>

    <br/>
    <apex:form>
        <apex:commandButton id="go" action="{!go}" value="Rerender" rerender="output"/>
        <apex:commandButton id="remove" action="{!remove}" value="Remove Key" rerender="output"/>
    </apex:form>
</apex:page>
```
SEE ALSO:
  
  * Apex Developer Guide: Platform Cache

## Cache Exceptions

The Cache namespace contains exception classes.

All exception classes support built-in methods for returning the error message and exception type. See Exception Class and Built-In Exceptions on page 3076 in the Apex Developer Guide.

The Cache namespace contains these exceptions.

<table>
<thead>
<tr>
<th>Exception</th>
<th>Thrown when</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cache.Session.SessionCacheException</td>
<td>An error occurred while adding or retrieving a value in the session cache.</td>
</tr>
<tr>
<td>Cache.Session.SessionCacheNoSessionException</td>
<td>An attempt is made to access the cache when the session cache isn’t available.</td>
</tr>
<tr>
<td>Cache.Org.OrgCacheException</td>
<td>An attempt is made to access a partition that doesn’t exist or whose name is invalid.</td>
</tr>
<tr>
<td>Cache.InvalidParamException</td>
<td>An invalid parameter value is passed into a method of Cache.Session or Cache.Org. This error occurs when:</td>
</tr>
<tr>
<td></td>
<td>• The key referenced is null or empty or is not alphanumeric.</td>
</tr>
<tr>
<td></td>
<td>• The key isn’t qualified with the namespace and partition in the format &lt;namespace&gt;.&lt;partition&gt;.&lt;key&gt;.</td>
</tr>
<tr>
<td></td>
<td>• The key isn’t qualified in the format &lt;key&gt; for the default partition, or for a key inserted through the partition object.</td>
</tr>
<tr>
<td></td>
<td>• The namespace referenced is null or empty.</td>
</tr>
<tr>
<td></td>
<td>• The partition name is null or empty or is not alphanumeric.</td>
</tr>
<tr>
<td></td>
<td>• Another referenced value is null.</td>
</tr>
<tr>
<td>Cache.ItemSizeLimitExceededException</td>
<td>A cache put call is made with an item that exceeds the maximum size limit. To fix this error, break the item into multiple, smaller items.</td>
</tr>
<tr>
<td>Cache.BulkApiKeysLimitExceededException</td>
<td>The number of key parameters passed into a bulk method - get(keys) or contains(setOfKeys) - exceeds the maximum limit of 10.</td>
</tr>
<tr>
<td>Cache.PlatformCacheInvalidOperationException</td>
<td>A cache put or remove call is made that is not allowed. For example, when calling put or remove inside a Visualforce constructor.</td>
</tr>
<tr>
<td>Cache.CacheBuilderExecutionException</td>
<td>This error occurs when the execution of the CacheBuilder fails; this could be due to an error in parsing, a permissions error while accessing records, or an issue with Apex callouts.</td>
</tr>
</tbody>
</table>
Visibility Enum

Use the Cache.Visibility enumeration in the Cache.Session or Cache.Org methods to indicate whether a cached value is visible only in the value’s namespace or in all namespaces.

Enum Values

The following are the values of the Cache.Visibility enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALL</td>
<td>The cached value is available to Apex code executing from any namespace. This is the default state.</td>
</tr>
<tr>
<td>NAMESPACE</td>
<td>The cached value is available to Apex code executing from the same namespace. If a key has the Visibility.NAMESPACE attribute, a get method initiated from a different namespace returns null.</td>
</tr>
</tbody>
</table>

Canvas Namespace

The Canvas namespace provides an interface and classes for canvas apps in Salesforce. The following are the interfaces and classes in the Canvas namespace.

IN THIS SECTION:

- ApplicationContext Interface
  Use this interface to retrieve application context information, such as the application version or URL.

- CanvasLifecycleHandler Interface
  Implement this interface to control context information and add custom behavior during the application render phase.

- ContextTypeEnum Enum
  Describes context data that can be excluded from canvas app context data. You specify which context types to exclude in the excludeContextTypes() method in your CanvasLifecycleHandler implementation.

- EnvironmentContext Interface
  Use this interface to retrieve environment context information, such as the app display location or the configuration parameters.
RenderContext Interface
A wrapper interface that is used to retrieve application and environment context information.

TestClass
Contains methods for automated testing of your Canvas classes.

Canvas Exceptions
The Canvas namespace contains exception classes.

ApplicationContext Interface
Use this interface to retrieve application context information, such as the application version or URL.

Namespace
Canvas

Usage
The ApplicationContext interface provides methods to retrieve application information about the canvas app that's being rendered. Most of the methods are read-only. For this interface, you don't need to create an implementation. Use the default implementation that Salesforce provides.

IN THIS SECTION:
ApplicationContext Methods

ApplicationContext Methods
The following are methods for ApplicationContext.

IN THIS SECTION:
getCanvasUrl() Retrieves the fully qualified URL of the canvas app.
getDeveloperName() Retrieves the internal API name of the canvas app.
getName() Retrieves the name of the canvas app.
getNamespace() Retrieves the namespace prefix of the canvas app.
getVersion() Retrieves the current version of the canvas app.
setCanvasUrlPath(newPath) Overrides the URL of the canvas app for the current request.
getCanvasUrl()
Retrieves the fully qualified URL of the canvas app.

Signature
public String getCanvasUrl()

Return Value
Type: String

Usage
Use this method to get the URL of the canvas app, for example:

getDeveloperName()
Retrieves the internal API name of the canvas app.

Signature
public String getDeveloperName()

Return Value
Type: String

Usage
Use this method to get the API name of the canvas app. You specify this value in the API Name field when you expose the canvas app by creating a connected app.

getName()
Retrieves the name of the canvas app.

Signature
public String getName()

Return Value
Type: String

Usage
Use this method to get the name of the canvas app.
getNamespace()
Retrieves the namespace prefix of the canvas app.

Signature
public String getNamespace()

Return Value
Type: String

Usage
Use this method to get the Salesforce namespace prefix that's associated with the canvas app.

getVersion()
Retrieves the current version of the canvas app.

Signature
public String getVersion()

Return Value
Type: String

Usage
Use this method to get the current version of the canvas app. This value changes after you update and republish a canvas app in an organization. If you are in a Developer Edition organization, using this method always returns the latest version.

setCanvasUrlPath(newPath)
Overrides the URL of the canvas app for the current request.

Signature
public void setCanvasUrlPath(String newPath)

Parameters
newPath
Type: String
The URL (not including domain) that you need to use to override the canvas app URL.

Return Value
Type: Void
Usage

Use this method to override the URL path and query string of the canvas app. Do not provide a fully qualified URL, because the provided URL string will be appended to the original canvas URL domain.

For example, if the current canvas app URL is https://myserver.com:6000/myAppPath and you call setCanvasUrlPath('/alternatePath/args?arg1=1&arg2=2'), the adjusted canvas app URL will be https://myserver.com:6000/alternatePath/args?arg1=1&arg2=2.

If the provided path results in a malformed URL, or a URL that exceeds 2,048 characters, a System.CanvasException will be thrown.

This method overrides the canvas app URL for the current request and does not permanently change the canvas app URL as configured in the UI for the Salesforce canvas app settings.

CanvasLifecycleHandler Interface

Implement this interface to control context information and add custom behavior during the application render phase.

Namespace

Canvas

Usage

Use this interface to specify what canvas context information is provided to your app by implementing the excludeContextTypes() method. Use this interface to call custom code when the app is rendered by implementing the onRender() method.

If you provide an implementation of this interface, you must implement excludeContextTypes() and onRender().

Example Implementation

The following example shows a simple implementation of CanvasLifecycleHandler that specifies that organization context information will be excluded and prints a debug message when the app is rendered.

```java
public class MyCanvasListener implements Canvas.CanvasLifecycleHandler{
    public Set<Canvas.ContextTypeEnum> excludeContextTypes(){
        Set<Canvas.ContextTypeEnum> excluded = new Set<Canvas.ContextTypeEnum>();
        excluded.add(Canvas.ContextTypeEnum.ORGANIZATION);
        return excluded;
    }

    public void onRender(Canvas.RenderContext renderContext){
        System.debug('Canvas lifecycle called.');
    }
}
```

IN THIS SECTION:

CanvasLifecycleHandler Methods
CanvasLifecycleHandler Methods

The following are methods for CanvasLifecycleHandler.

IN THIS SECTION:

excludeContextTypes()
Lets the implementation exclude parts of the CanvasRequest context, if the application does not need it.

onRender(renderContext)
Invoked when a canvas app is rendered. Provides the ability to set and retrieve canvas application and environment context information during the application render phase.

excludeContextTypes()
Lets the implementation exclude parts of the CanvasRequest context, if the application does not need it.

Signature

public Set<Canvas.ContextTypeEnum> excludeContextTypes()

Return Value

Type: SET<Canvas.ContextTypeEnum>

This method must return null or a set of zero or more ContextTypeEnum values. Returning null enables all attributes by default. ContextTypeEnum values that can be set are:

- Canvas.ContextTypeEnum.ORGANIZATION
- Canvas.ContextTypeEnum.RECORD_DETAIL
- Canvas.ContextTypeEnum.USER

See ContextTypeEnum on page 238 for more details on these values.

Usage

Implement this method to specify which attributes to disable in the context of the canvas app. A disabled attribute will set the associated canvas context information to null.

Disabling attributes can help improve performance by reducing the size of the signed request and canvas context. Also, disabled attributes do not need to be retrieved by Salesforce, which further improves performance.

See the Canvas Developer Guide for more information on context information in the Context object that’s provided in the CanvasRequest.

Example

This example implementation specifies that the organization information will be disabled in the canvas context.

```java
public Set<Canvas.ContextTypeEnum> excludeContextTypes() {
    Set<Canvas.ContextTypeEnum> excluded = new Set<Canvas.ContextTypeEnum>();
    excluded.add(Canvas.ContextTypeEnum.ORGANIZATION);
    return excluded;
}
```
onRender(renderContext)
Invoked when a canvas app is rendered. Provides the ability to set and retrieve canvas application and environment context information during the application render phase.

Signature
public void onRender(Canvas.RenderContext renderContext)

Parameters
renderContext
Type: Canvas.RenderContext

Return Value
Type: Void

Usage
If implemented, this method is called whenever the canvas app is rendered. The implementation can set and retrieve context information by using the provided Canvas.RenderContext.
This method is called whenever signed request or context information is retrieved by the client. See the Canvas Developer Guide for more information on signed request authentication.

Example
This example implementation prints 'Canvas lifecycle called.' to the debug log when the canvas app is rendered.

```java
public void onRender(Canvas.RenderContext renderContext) {
    System.debug('Canvas lifecycle called. ');
}
```

ContextTypeEnum Enum
Describes context data that can be excluded from canvas app context data. You specify which context types to exclude in the excludeContextTypes() method in your CanvasLifecycleHandler implementation.

Namespace
Canvas

Enum Values

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORGANIZATION</td>
<td>Exclude context information about the organization in which the canvas app is running.</td>
</tr>
<tr>
<td>Value</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>RECORD_DETAIL</td>
<td>Exclude context information about the object record on which the canvas app appears.</td>
</tr>
<tr>
<td>USER</td>
<td>Exclude context information about the current user.</td>
</tr>
</tbody>
</table>

**EnvironmentContext Interface**

Use this interface to retrieve environment context information, such as the app display location or the configuration parameters.

**Namespace**

Canvas

**Usage**

The `EnvironmentContext` interface provides methods to retrieve environment information about the current canvas app. For this interface, you don’t need to create an implementation. Use the default implementation that Salesforce provides.

**EnvironmentContext Methods**

The following are methods for `EnvironmentContext`.

**IN THIS SECTION:**

- `addEntityField(fieldName)`
  Adds a field to the list of object fields that are returned in the signed request Record object when the component appears on a Visualforce page that’s placed on an object.
- `addEntityFields(fieldNames)`
  Adds a set of fields to the list of object fields that are returned in the signed request Record object when the component appears on a Visualforce page that’s placed on an object.
- `getDisplayLocation()`
  Retrieves the display location where the canvas app is being called from. For example, a value of Visualforce page.
- `getEntityFields()`
  Retrieves the list of object fields that are returned in the signed request Record object when the component appears on a Visualforce page that’s placed on an object.
- `getLocationUrl()`
  Retrieves the location URL of the canvas app.
- `getParametersAsJSON()`
  Retrieves the current custom parameters for the canvas app. Parameters are returned as a JSON string.
getSublocation()
Retrieves the display sublocation where the canvas app is being called from.

setParametersAsJSON(jsonString)
Sets the custom parameters for the canvas app.

addEntityField(fieldName)
Adds a field to the list of object fields that are returned in the signed request Record object when the component appears on a Visualforce page that’s placed on an object.

Signature
public void addEntityField(String fieldName)

Parameters

<table>
<thead>
<tr>
<th>fieldName</th>
<th>Type: String</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The object field name that you need to add to the list of returned fields. Using ‘*’ adds all fields that the user has permission to view.</td>
</tr>
</tbody>
</table>

Return Value
Type: Void

Usage
When you use the <apex:canvasApp> component to display a canvas app on a Visualforce page, and that page is associated with an object (placed on the page layout, for example), you can specify fields to be returned from the related object. See the Canvas Developer Guide for more information on the Record object.

Use addEntityField() to add a field to the list of object fields that are returned in the signed request Record object. By default the list of fields includes ID. You can add fields by name or add all fields that the user has permission to view by calling addEntityField('*').

You can inspect the configured list of fields by using Canvas.EnvironmentContext.getEntityFields().

Example
This example adds the Name and BillingAddress fields to the list of object fields. This example assumes the canvas app will appear in a Visualforce page that’s associated with the Account page layout.

```java
Canvas.EnvironmentContext env = renderContext.getEnvironmentContext();
// Add Name and BillingAddress to fields (assumes we'll run from the Account detail page)
env.addEntityField('Name');
env.addEntityField('BillingAddress');
```

addEntityFields(fieldNames)
Adds a set of fields to the list of object fields that are returned in the signed request Record object when the component appears on a Visualforce page that’s placed on an object.
public void addEntityFields(Set<String> fieldNames)

Parameters

fieldNames
Type: SET<String>
The set of object field names that you need to add to the list of returned fields. If an item in the set is ‘*’, all fields that the user has permission to view are added.

Return Value

Type: Void

Usage

When you use the <apex:canvasApp> component to display a canvas app on a Visualforce page, and that page is associated with an object (placed on the page layout, for example), you can specify fields to be returned from the related object. See the Canvas Developer Guide for more information on the Record object.

Use addEntityFields() to add a set of one or more fields to the list of object fields that are returned in the signed request Record object. By default the list of fields includes ID. You can add fields by name or add all fields that the user has permission to view by adding a set that includes ‘*’ as one of the strings.

You can inspect the configured list of fields by using Canvas.EnvironmentContext.getEntityFields().

Example

This example adds the Name, BillingAddress, and YearStarted fields to the list of object fields. This example assumes that the canvas app will appear in a Visualforce page that’s associated with the Account page layout.

```java
Canvas.EnvironmentContext env = renderContext.getEnvironmentContext();

// Add Name, BillingAddress and YearStarted to fields (assumes we'll run from the Account detail page)
Set<String> fields = new Set<String>{'Name','BillingAddress','YearStarted'};
env.addEntityFields(fields);
```

getDisplayLocation()

Retrieves the display location where the canvas app is being called from. For example, a value of Visualforce page.

Signature

public String getDisplayLocation()
• ChatterFeed—The canvas app was called from a Chatter canvas feed item.
• MobileNav—The canvas app was called from the navigation menu.
• OpenCTI—The canvas app was called from an Open CTI component.
• PageLayout—The canvas app was called from an element within a page layout. If the displayLocation is PageLayout, one of the subLocation values might be returned.
• Publisher—The canvas app was called from a canvas custom quick action.
• ServiceDesk—The canvas app was called from a Salesforce Console component.
• Visualforce—The canvas app was called from a Visualforce page.
• None—The canvas app was called from the Canvas App Previewer.

Usage

Use this method to obtain the display location for the canvas app.

getEntityFields()

Retrieves the list of object fields that are returned in the signed request Record object when the component appears on a Visualforce page that’s placed on an object.

Signature

public List<String> getEntityFields()

Return Value

Type: LIST<String>

Usage

When you use the <apex:canvasApp> component to display a canvas app on a Visualforce page, and that page is associated with an object (placed on the page layout, for example), you can specify fields to be returned from the related object. See the Canvas Developer Guide for more information on the Record object.

Use getEntityFields() to retrieve the list of object fields that are returned in the signed request Record object. By default the list of fields includes ID. The list of fields can be configured by using the Canvas.EnvironmentContext.addEntityField(fieldName) or Canvas.EnvironmentContext.addEntityFields(fieldNames) methods.

Example

This example gets the current list of object fields and retrieves each item in the list, printing each field name to the debug log.

```
Canvas.EnvironmentContext env = renderContext.getEnvironmentContext();
List<String> entityFields = env.getEntityFields();
for (String fieldVal : entityFields) {
    System.debug('Environment Context entityField: ' + fieldVal);
}
```

If the canvas app that’s using this lifecycle code was run from the detail page of an Account, the debug log output might look like:

```
Environment Context entityField: Id
```
getLocationUrl()
Retrieves the location URL of the canvas app.

Signature
public String getLocationUrl()

Return Value
Type: String

Usage
Use this method to obtain the URL of the page where the user accessed the canvas app. For example, if the user accessed your app by clicking a link on the Chatter tab, this method returns the URL of the Chatter tab, which would be similar to ‘https://MyDomainName.my.salesforce.com/_ui/core/chatter/ui/ChatterPage’.

getParametersAsJSON()
Retrieves the current custom parameters for the canvas app. Parameters are returned as a JSON string.

Signature
public String getParametersAsJSON()

Return Value
Type: String

Usage
Use this method to get the current custom parameters for the canvas app. The parameters are returned in a JSON string that can be de-serialized by using the System.JSON.deserializeUntyped(jsonString) method.

Custom parameters can be modified by using the Canvas.EnvironmentContext.setParametersAsJSON(jsonString) method.

Example
This example gets the current custom parameters, de-serializes them into a map, and prints the results to the debug log.

```java
Canvas.EnvironmentContext env = renderContext.getEnvironmentContext();

// Get current custom params
Map<String, Object> currentParams =
    (Map<String, Object>) JSON.deserializeUntyped(env.getParametersAsJSON());
System.debug('Environment Context custom parameters: ' + currentParams);
```

getSublocation()
Retrieves the display sublocation where the canvas app is being called from.
Signature

public String getSublocation()

Return Value

Type: String

The return value can be one of the following strings:

• `S1MobileCardFullview`—The canvas app was called from a mobile card.
• `S1MobileCardPreview`—The canvas app was called from a mobile card preview. The user must click the preview to open the app.
• `S1RecordHomePreview`—The canvas app was called from a record detail page preview. The user must click the preview to open the app.
• `S1RecordHomeFullview`—The canvas app was called from a page layout.

Usage

Use this method to obtain the display sublocation for the canvas app. Use only if the primary display location can be displayed on mobile devices.

`setParametersAsJSON(jsonString)`

Sets the custom parameters for the canvas app.

Signature

public void setParametersAsJSON(String jsonString)

Parameters

`jsonString`

Type: String

The custom parameters that you need to set, serialized into a JSON format string.

Return Value

Type: Void

Usage

Use this method to set the current custom parameters for the canvas app. The parameters must be provided in a JSON string. You can use the `System.JSON.serialize(objectToSerialize)` method to serialize a map into a JSON string.

Setting the custom parameters will overwrite the custom parameters that are set for the current request. If you need to modify the current custom parameters, first get the current set of custom parameters by using `getParametersAsJSON()`, modify the retrieved parameter set as needed, and then use this modified set in your call to `setParametersAsJSON()`.

If the provided JSON string exceeds 32KB, a `System.CanvasException` will be thrown.
Example

This example gets the current custom parameters, adds a new `newCustomParam` parameter with a value of 'TESTVALUE', and sets the current custom parameters.

```java
Canvas.EnvironmentContext env = renderContext.getEnvironmentContext();

// Get current custom params
Map<String, Object> previousParams =
    (Map<String, Object>) JSON.deserializeUntyped(env.getParametersAsJSON());

// Add a new custom param
previousParams.put('newCustomParam', 'TESTVALUE');

// Now replace the parameters with the current parameters plus our new custom param
env.setParametersAsJSON(JSON.serialize(previousParams));
```

RenderContext Interface

A wrapper interface that is used to retrieve application and environment context information.

Namespace

Canvas

Usage

Use this interface to retrieve application and environment context information for your canvas app. For this interface, you don't need to create an implementation. Use the default implementation that Salesforce provides.

IN THIS SECTION:

RenderContext Methods

RenderContext Methods

The following are methods for `RenderContext`.

IN THIS SECTION:

getApplicationContext()

Retrieves the application context information.

getEnvironmentContext()

Retrieves the environment context information.

getApplicationContext()

Retrieves the application context information.
Signature

public Canvas.ApplicationContext getApplicationContext()

Return Value

Type: Canvas.ApplicationContext

Usage

Use this method to get the application context information for your canvas app.

Example

The following example implementation of the CanvasLifecycleHandler onRender() method uses the provided RenderContext to retrieve the application context information and then checks the namespace, version, and app URL.

```java
public void onRender(Canvas.RenderContext renderContext){
    Canvas.ApplicationContext app = renderContext.getApplicationContext();
    if (!'MyNamespace'.equals(app.getNamespace())){
        // This application is installed, add code as needed
        ...
    }

    // Check the application version
    Double currentVersion = Double.valueOf(app.getVersion());
    if (currentVersion <= 5){
        // Add version specific code as needed
        ...
        // Tell the canvas application to operate in deprecated mode
        app.setCanvasUrlPath('/canvas?deprecated=true');
    }
}
```

getEnvironmentContext()

Retrieves the environment context information.

Signature

public Canvas.EnvironmentContext getEnvironmentContext()

Return Value

Type: Canvas.EnvironmentContext

Usage

Use this method to get the environment context information for your canvas app.
Example

The following example implementation of the CanvasLifecycleHandler onRender() method uses the provided RenderContext to retrieve the environment context information and then modifies the custom parameters.

```java
public void onRender(Canvas.RenderContext renderContext) {
    Canvas.EnvironmentContext env = renderContext.getEnvironmentContext();

    // Retrieve the custom params
    Map<String, Object> previousParams = (Map<String, Object>)
        JSON.deserializeUntyped(env.getParametersAsJSON());
    previousParams.put('param1', 1);
    previousParams.put('param2', 3.14159);
    ...

    // Now, add in some opportunity record IDs
    Opportunity[] o = [select id, name from opportunity];
    previousParams.put('opportunities', o);

    // Now, replace the parameters
    env.setParametersAsJSON(JSON.serialize(previousParams));
}
```

Test Class

Contains methods for automated testing of your Canvas classes.

Namespace

Canvas

Usage

Use this class to test your implementation of Canvas.CanvasLifecycleHandler with mock test data. You can create a test Canvas.RenderContext with mock application and environment context data and use this data to verify that your CanvasLifecycleHandler is being invoked correctly.

IN THIS SECTION:

Test Constants
The Test class provides constants that are used as keys when you set mock application and environment context data.

Test Methods
The Test class provides methods for creating test contexts and invoking your CanvasLifecycleHandler with mock data.

Test Constants
The Test class provides constants that are used as keys when you set mock application and environment context data.
When you call `Canvas.Test.mockRenderContext(applicationContextTestValues, environmentContextTestValues)`, you need to provide maps of key-value pairs to represent your mock application and environment context data. The Test class provides static constant strings that you can use as keys for various parts of the application and environment context.

<table>
<thead>
<tr>
<th>Constant</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>KEY_CANVAS_URL</td>
<td>Represents the canvas app URL key in the ApplicationContext.</td>
</tr>
<tr>
<td>KEY_DEVELOPER_NAME</td>
<td>Represents the canvas app developer or API name key in the ApplicationContext.</td>
</tr>
<tr>
<td>KEY_DISPLAY_LOCATION</td>
<td>Represents the canvas app display location key in the EnvironmentContext.</td>
</tr>
<tr>
<td>KEY_LOCATION_URL</td>
<td>Represents the canvas app location URL key in the EnvironmentContext.</td>
</tr>
<tr>
<td>KEY_NAME</td>
<td>Represents the canvas app name key in the ApplicationContext.</td>
</tr>
<tr>
<td>KEY_NAMESPACE</td>
<td>Represents the canvas app namespace key in the ApplicationContext.</td>
</tr>
<tr>
<td>KEY_SUB_LOCATION</td>
<td>Represents the canvas app sublocation key in the EnvironmentContext.</td>
</tr>
<tr>
<td>KEY_VERSION</td>
<td>Represents the canvas app version key in the ApplicationContext.</td>
</tr>
</tbody>
</table>

Test Methods

The Test class provides methods for creating test contexts and invoking your CanvasLifecycleHandler with mock data. The following are methods for Test. All are static methods.

**IN THIS SECTION:**

- `mockRenderContext(applicationContextTestValues, environmentContextTestValues)`
  Creates and returns a test Canvas.RenderContext based on the provided application and environment context parameters.
- `testCanvasLifecycle(lifecycleHandler, mockRenderContext)`
  Calls the canvas test framework to invoke a CanvasLifecycleHandler with the provided RenderContext.

**mockRenderContext(applicationContextTestValues, environmentContextTestValues)**

Creates and returns a test Canvas.RenderContext based on the provided application and environment context parameters.

**Signature**

```java
public static Canvas.RenderContext mockRenderContext(Map<String, String> applicationContextTestValues, Map<String, String> environmentContextTestValues)
```

**Parameters**

- `applicationContextTestValues`
  Type: Map<String, String>
  Specifies a map of key-value pairs that provide mock application context data. Use constants that are provided by Canvas.Test as keys. If `null` is provided for this parameter, the canvas framework generates some default mock application context values.
environmentContextTestValues

- **Type:** `Map<String,String>`

  Specifies a map of key-value pairs that provide mock environment context data. Use constants provided by Canvas.Test as keys. If `null` is provided for this parameter, the canvas framework generates some default mock environment context values.

**Return Value**

- **Type:** `Canvas.RenderContext`

**Usage**

Use this method to create a mock `Canvas.RenderContext`. Use the returned RenderContext in calls to `Canvas.Test.testCanvasLifecycle(lifecycleHandler, mockRenderContext)` for testing `Canvas.CanvasLifecycleHandler` implementations.

**Example**

The following example creates maps to represent mock application and environment context data and generates a test `Canvas.RenderContext`. This test RenderContext can be used in a call to `Canvas.Test.testCanvasLifecycle(lifecycleHandler, mockRenderContext)`.

```java
Map<String,String> appValues = new Map<String,String>();
appValues.put(Canvas.Test.KEY_NAMESPACE,'alternateNamespace');
appValues.put(Canvas.Test.KEY_VERSION,'3.0');

Map<String,String> envValues = new Map<String,String>();
envValues.put(Canvas.Test.KEY_DISPLAY_LOCATION,'Chatter');
envValues.put(Canvas.Test.KEY_LOCATION_URL,'https://MyDomainName.my.salesforce.com/_ui/core/chatter/ui/ChatterPage');

Canvas.RenderContext mock = Canvas.Test.mockRenderContext(appValues,envValues);
```

**testCanvasLifecycle(lifecycleHandler, mockRenderContext)**

Calls the canvas test framework to invoke a `CanvasLifecycleHandler` with the provided RenderContext.

**Signature**

```java
public static Void testCanvasLifecycle(Canvas.CanvasLifecycleHandler lifecycleHandler,Canvas.RenderContext mockRenderContext)
```

**Parameters**

- **lifecycleHandler**
  - **Type:** `Canvas.CanvasLifecycleHandler`
  - Specifies the `CanvasLifecycleHandler` implementation that you need to invoke.

- **mockRenderContext**
  - **Type:** `Canvas.RenderContext`
  - Specifies the RenderContext information that you need to provide to the invoked `CanvasLifecycleHandler`. If `null` is provided for this parameter, the canvas framework generates and uses a default mock RenderContext.
Return Value
Type: Void

Usage
Use this method to invoke an implementation of `Canvas.CanvasLifecycleHandler.onRender(renderContext)` with a mock `Canvas.RenderContext` that you provide.

Example
The following example creates maps to represent mock application and environment context data and generates a test `Canvas.RenderContext`. This test `RenderContext` is then used to invoke a `Canvas.CanvasLifecycleHandler`.

```java
// Set some application context data in a Map
Map<String, String> appValues = new Map<String, String>();
appValues.put(Canvas.Test.KEY_NAMESPACE, 'alternateNamespace');
appValues.put(Canvas.Test.KEY_VERSION, '3.0');

// Set some environment context data in a Map
Map<String, String> envValues = new Map<String, String>();
envValues.put(Canvas.Test.KEY_DISPLAY_LOCATION, 'Chatter');
envValues.put(Canvas.Test.KEY_LOCATION_URL, 'https://MyDomainName.my.salesforce.com/_ui/core/chatter/ui/ChatterPage');

// Create a mock RenderContext using the test application and environment context data
Canvas.RenderContext mock = Canvas.Test.mockRenderContext(appValues, envValues);

// Set some custom params on the mock RenderContext
mock.getEnvironmentContext().setParametersAsJSON('{"param1":1,"boolParam":true,"stringParam":"test string"}');

// Use the mock RenderContext to invoke a CanvasLifecycleHandler
Canvas.Test.testCanvasLifecycle(handler, mock)
```

Canvas Exceptions
The `Canvas` namespace contains exception classes.

All exception classes support built-in methods for returning the error message and exception type. See Exception Class and Built-In Exceptions.

The `Canvas` namespace contains this exception:

<table>
<thead>
<tr>
<th>Exception</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Canvas.CanvasRenderException</code></td>
<td>Use this class in your implementation of <code>Canvas.CanvasLifecycleHandler.onRender(renderContext)</code>. To show an error to the user in your <code>onRender()</code> implementation, throw a <code>Canvas.CanvasRenderException</code>, and the canvas framework will render the error message to the user. This exception will be managed only within the <code>onRender()</code> method.</td>
</tr>
</tbody>
</table>
Example

The following example implementation of `onRender()` catches a `CanvasException` that was thrown because a canvas URL was set with a string that exceeded the maximum length. A `CanvasRenderException` is created and thrown to display the error to the user.

```java
public class MyCanvasListener
    implements Canvas.CanvasLifecycleHandler {

    public void onRender(Canvas.RenderContext renderContext) {
        Canvas.ApplicationContext app = renderContext.getApplicationContext();

        // Code to generate a URL string that is too long
        // ...

        // Try to set the canvas app URL using the invalid URL string
        try {
            app.setCanvasUrlPath(aUrlPathThatIsTooLong);
        } catch (CanvasException e) {
            // Display error to user by throwing a new CanvasRenderException
            throw new Canvas.CanvasRenderException(e.getMessage());
        }
    }
}
```

See the Canvas Developer Guide for additional examples that use `CanvasRenderException`.

ChatterAnswers Namespace

The ChatterAnswers namespace provides an interface for creating Account records.

The following is the interface in the ChatterAnswers namespace.

IN THIS SECTION:

AccountCreator Interface

Creates Account records that will be associated with Chatter Answers users.

AccountCreator Interface

Creates Account records that will be associated with Chatter Answers users.

Namespace

ChatterAnswers

Usage

The ChatterAnswers.AccountCreator is specified in the `registrationClassName` attribute of a chatteranswers:registration Visualforce component. This interface is called by Chatter Answers and allows for custom creation of Account records used for portal users.
To implement the `ChatterAnswers.AccountCreator` interface, you must first declare a class with the `implements` keyword as follows:

```java
public class ChatterAnswersRegistration implements ChatterAnswers.AccountCreator {

Next, your class must provide an implementation for the following method:

```java
public String createAccount(String firstname, String lastname, Id siteAdminId) {
    // Your code here
}
```

The implemented method must be declared as `global` or `public`.

IN THIS SECTION:

AccountCreator Methods

AccountCreator Example Implementation

**AccountCreator Methods**

The following are methods for `AccountCreator`.

IN THIS SECTION:

`createAccount(firstName, lastName, siteAdminId)`

Accepts basic user information and creates an Account record. The implementation of this method returns the account ID.

**createAccount(firstName, lastName, siteAdminId)**

Accepts basic user information and creates an Account record. The implementation of this method returns the account ID.

**Signature**

```java
public String createAccount(String firstName, String lastName, Id siteAdminId)
```

**Parameters**

- `firstName`
  - Type: `String`
  - The first name of the user who is registering.

- `lastName`
  - Type: `String`
  - The last name of the user who is registering.

- `siteAdminId`
  - Type: `ID`
  - The user ID of the Site administrator, used for notification if any exceptions occur.

**Return Value**

- Type: `String`
AccountCreator Example Implementation

This is an example implementation of the ChatterAnswers.AccountCreator interface. The createAccount method implementation accepts user information and creates an Account record. The method returns a String value for the Account ID.

```java
public class ChatterAnswersRegistration implements ChatterAnswers.AccountCreator {
    public String createAccount(String firstname, String lastname, Id siteAdminId) {
        Account a = new Account(name = firstname + ' ' + lastname, ownerId = siteAdminId);

        insert a;
        return a.Id;
    }
}
```

This example tests the code above.

```java
@isTest
private class ChatterAnswersCreateAccountTest {
    static testMethod void validateAccountCreation() {
        User[] user = [SELECT Id, Firstname, Lastname from User WHERE UserType='Standard'];

        if (user.size() == 0) { return; }
        String firstName = user[0].FirstName;
        String lastName = user[0].LastName;
        String userId = user[0].Id;
        String accountId = new ChatterAnswersRegistration().createAccount(firstName, lastName, userId);
        Account acct = [SELECT name, ownerId from Account where Id =: accountId];
        System.assertEquals(firstName + ' ' + lastName, acct.name);
        System.assertEquals(userId, acct.ownerId);
    }
}
```

CommercePayments Namespace

Use the CommercePayments namespace to provide a safe and customizable platform for managing customer payments and refunds.

To review CommercePayments use cases and walkthroughs, go to Use Cases for the CommercePayments Namespace.

The following are the classes in the CommercePayments namespace.

IN THIS SECTION:

- **AbstractResponse Class**
  Contains the normalized response fields from payment gateways that are common to all the other gateway responses.

- **AddressRequest Class**
  Contains address request data that is sent to a gateway adapter during a service call.

- **AuditParamsRequest**
  AuditParamsRequest is used for audit parameters in a transaction request. This is an abstract request class that is extended by the BaseRequest class.
AbstractTransactionResponse Class
Abstract class for storing normalized information sent from payment gateways about a payment transaction. Holds the common response fields sent from payment gateways for authorization, sale, capture, and refund transactions.

AuthApiPaymentMethodRequest Class
Sends information about a payment method to a gateway adapter during an authorization service call.

AuthorizationReversalRequest Class
Sends information about an authorization reversal request to a gateway adapter during a service call.

AuthorizationReversalResponse Class
Response sent by the payment gateway following a payment authorization reversal service.

AuthorizationRequest Class
Sends information about an authorization request to a gateway adapter during a service call.

AuthorizationResponse Class
Response sent by the payment gateway adapter for an authorization service.

BaseApiPaymentMethodRequest Class
Abstract class used to send information about a payment method to a gateway adapter during a service call.

BaseNotification Class
Abstract class for storing notification information sent from payment gateways.

BasePaymentMethodRequest Class
Abstract class for storing information about payment methods.

BaseRequest Class
BaseRequest is extended by all the request classes.

CaptureNotification Class
When a payment gateway sends a notification for a capture transaction, the payment gateway adapter creates the CaptureNotification object to store information about the notification.

CaptureRequest Class
Represents a capture request. This class extends the BaseRequest class and inherits all its methods.

CaptureResponse Class
The payment gateway adapter sends this response for the capture request type. This class extends AbstractResponse and inherits its methods.

CardCategory Enum
Defines whether the payment method represents a credit card or a debit card.

CardPaymentMethodRequest Class
Sends data related to a card payment method to a gateway adapter during a service call.

CustomMetadataTypeInfo Class
Access information about custom metadata. The PaymentGatewayAdapter can send CustomMetadataTypeInfo to transaction requests through the response object's SalesforceResultCodeInfo.

GatewayErrorResponse Class
Use to respond with an error indication following errors from the PaymentGateway adapter, such as request-forbidden responses, custom validation errors, or expired API tokens.
GatewayNotificationResponse Class
When the payment gateway sends a notification to the payments platform, the platform responds with a GatewayNotificationResponse indicating whether the platform succeeded or failed at receiving the notification.

GatewayResponse Interface
Generic payment gateway response interface. This class extends the CaptureResponse on page 309, AbstractTransactionResponse on page 265, and AbstractResponse on page 256 classes and inherits all their properties. It has no unique methods or parameters.

NotificationClient Class
Communicates with the payment platform regarding the gateway’s notification.

NotificationSaveResult Class
Contains the result of the payment platform’s attempt to record data from the gateway’s notification.

NotificationStatus Enum
Shows whether the payments platform successfully received the notification from the gateway.

PaymentGatewayAdapter Interface
PaymentGatewayAdapters can implement this interface in order to process requests.

PaymentGatewayAsyncAdapter Interface
Implement the interface to allow customers to process payments asynchronously.

PaymentGatewayContext Class
Wraps the information related to a payment request.

PaymentGatewayNotificationContext Class
Wraps the information related to a gateway notification.

PaymentMethodTokenizationRequest Class
Stores data about a request to tokenize a card payment method. The tokenization process occurs in the payment gateway. This process replaces sensitive customer data, such as a card number or CVV, with unique identification symbols. The symbols are used while the data is handled by Salesforce, the payment gateway, and the customer bank, allowing Salesforce to store the token without storing sensitive customer data.

PaymentMethodTokenizationResponse Class
Gateway response sent by payment gateway adapters for the payment method tokenization request. The response includes the payment method’s token ID value.

PaymentGatewayNotificationRequest Class
Contains the notification request data from the gateway.

PaymentsHttp Class
Makes an HTTP request to start the interaction with the payment gateway.

RefundRequest Class
Sends data related to a refund to the payment gateway adapter.

ReferencedRefundNotification Class
When a payment gateway sends a notification for a refund transaction, the payment gateway adapter creates the ReferencedRefundNotification object to store information about notification.

ReferencedRefundRequest
Access information about the referenced refund requests. Extends the RefundRequest class.

ReferencedRefundResponse Class
The payment gateway adapter sends this response for the ReferencedRefund request type.
RequestType Enum
Defines the type of payment transaction request made to the payment gateway.

SaleApiPaymentMethodRequest Class
Sends data related to a card payment method to a gateway adapter during a sale service call.

SaleRequest Class
Stores information about a sales request.

SaleResponse Class
Response sent by payment gateway adapters for a sales service.

SalesforceResultCode Enum
Defines the gateway call status values in Salesforce based on the call status values that the payment gateway returned.

SalesforceResultCodeInfo
Stores Salesforce result code information from payment gateway adapters.

AbstractResponse Class
Contains the normalized response fields from payment gateways that are common to all the other gateway responses.

Namespace
CommercePayments

Usage
You must specify the CommercePayments namespace when creating an instance of this class. The constructor of this class takes no arguments. For example:

```
```

This class can’t be instantiated on its own. This class implements the GatewayResponse class. Other GatewayResponse classes extend this class to inherit common properties.

IN THIS SECTION:
AbstractResponse Methods

AbstractResponse Methods
The following are methods for AbstractResponse.

IN THIS SECTION:

```
setGatewayAvsCode(gatewayAvsCode)
Sets the AVS (address verification system) result code information that the gateway returned. Maximum length of 64 characters.

setGatewayDate(gatewayDate)
Sets the date that the transaction occurred. Some gateways don’t send this value.

setGatewayMessage(gatewayMessage)
Sets error messages that the gateway returned for the payment request. Maximum length of 255 characters.
```
setGatewayResultCode(gatewayResultCode)
Sets a gateway-specific result code. The code may be mapped to a Salesforce-specific result code. Maximum length of 64 characters.

setGatewayResultCodeDescription(gatewayResultCodeDescription)
Sets a description of the gateway-specific result code that a payment gateway returned. Maximum length of 1000 characters.

setSalesforceResultCodeInfo(salesforceResultCodeInfo)
Sets the Salesforce-specific result code information. Payment gateways have many response codes for payment calls. Salesforce uses the result code information to map payment gateway codes to a predefined set of standard Salesforce result codes.

setGatewayAvsCode(gatewayAvsCode)
Sets the AVS (address verification system) result code information that the gateway returned. Maximum length of 64 characters.

Signature

global void setGatewayAvsCode(String gatewayAvsCode)

Parameters

gatewayAvsCode
  Type: String
  Code sent by gateways that use an address verification system.

Return Value

Type: void

setGatewayDate(gatewayDate)
Sets the date that the transaction occurred. Some gateways don’t send this value.

Signature

global void setGatewayDate(Datetime gatewayDate)

Parameters

gatewayDate
  Type: Datetime
  Date and time of the gateway communication.

Return Value

Type: void

setGatewayMessage(gatewayMessage)
Sets error messages that the gateway returned for the payment request. Maximum length of 255 characters.
Signature

`global void setGatewayMessage(String gatewayMessage)`

Parameters

gatewayMessage
   Type: String
   Information on error messages sent from the gateway.

Return Value

Type: void

`setGatewayResultCode(String gatewayResultCode)`

Sets a gateway-specific result code. The code may be mapped to a Salesforce-specific result code. Maximum length of 64 characters.

Signature

`global void setGatewayResultCode(String gatewayResultCode)`

Parameters

gatewayResultCode
   Type: String
   Gateway-specific result code. Must be used to map a Salesforce-specific result code.

Return Value

Type: void

`setGatewayResultCodeDescription(String gatewayResultCodeDescription)`

Sets a description of the gateway-specific result code that a payment gateway returned. Maximum length of 1000 characters.

Signature

`global void setGatewayResultCodeDescription(String gatewayResultCodeDescription)`

Parameters

gatewayResultCodeDescription
   Type: String
   Description of the gateway’s result code. Use this field to learn more about why the gateway returned a certain result code.

Return Value

Type: void
setSalesforceResultCodeInfo(salesforceResultCodeInfo)

Sets the Salesforce-specific result code information. Payment gateways have many response codes for payment calls. Salesforce uses the result code information to map payment gateway codes to a predefined set of standard Salesforce result codes.

Signature


Parameters

salesforceResultCodeInfo
Type: commercepayments.SalesforceResultCodeInfo on page 377
Description of the Salesforce result code value.

Return Value

Type: void

AddressRequest Class

Contains address request data that is sent to a gateway adapter during a service call.

Namespace

CommercePayments

Usage

Contains information about the payment method’s address. Use this information in authorization, sale, and tokenization requests. The payment gateway adapter uses information in an AddressRequest object to construct a JSON request to send to the payment gateway. The constructor of this class takes no arguments. For example:

CommercePayments.AddressRequest adr = new CommercePayments.AddressRequest();

IN THIS SECTION:

AddressRequest Constructors
AddressRequest Properties
AddressRequest Methods

AddressRequest Constructors

The following are constructors for AddressRequest.
IN THIS SECTION:

AddressRequest(street, city, state, country, postalCode)

Constructs a sample address. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

AddressRequest(street, city, state, country, postalCode)

Constructs a sample address. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

Signature

global AddressRequest(String street, String city, String state, String country, String postalCode)

Parameters

street
  Type: String
  Street for the payment method's address.

city
  Type: String
  City for the payment method's address.

state
  Type: String
  State for the payment method's address.

country
  Type: String
  Country for the payment method's address.

postalCode
  Type: String
  Postal code for the payment method's address.

AddressRequest Properties

The following are properties for AddressRequest.

IN THIS SECTION:

  city
  City of the payment method address.

  companyName
  Company name of the payment method address.

  country
  Country for the payment method address.
postalCode
Postal code for the payment method address.

state
State for the payment method address.

street
Street for the payment method address.

city
City of the payment method address.

Signature
global String city {get; set;}

Property Value
Type: String

companyName
Company name of the payment method address.

Signature
global String companyName {get; set;}

Property Value
Type: String

country
Country for the payment method address.

Signature
global String country {get; set;}

Property Value
Type: String

postalCode
Postal code for the payment method address.

Signature
global String postalCode {get; set;}

261
Property Value
Type: String

state
State for the payment method address.

Signature
global String state {get; set;}

Property Value
Type: String

street
Street for the payment method address.

Signature
global String street {get; set;}

Property Value
Type: String

AddressRequest Methods
The following are methods for AddressRequest.

IN THIS SECTION:
equals(obj)
Maintains the integrity of lists of type AddressRequest by determining the equality of external objects in a list. This method is dynamic and is based on the equals method in Java.

hashCode()
Maintains the integrity of lists of type AddressRequest.

toString()
Converts a date to a string.

equals(obj)
Maintains the integrity of lists of type AddressRequest by determining the equality of external objects in a list. This method is dynamic and is based on the equals method in Java.

Signature
global Boolean equals(Object obj)
Parameters

**obj**
Type: Object
External object whose key is to be validated.

Return Value
Type: Boolean

**hashCode()**
Maintains the integrity of lists of type AddressRequest.

Signature
`global Integer hashCode()`

Return Value
Type: Integer

**toString()**
Converts a date to a string.

Signature
`global String toString()`

Return Value
Type: String

**AuditParamsRequest**

AuditParamsRequest is used for audit parameters in a transaction request. This is an abstract request class that is extended by the BaseRequest class.

Namespace
CommercePayments

Usage
AuditParamsRequest is an abstract class that holds attributes related to audit parameters such as email, IP address, MAC address, and phone number. This class can't be instantiated on its own. All CommercePayments request classes extend this class.
AuditParamsRequest Constructors

The following are constructors for AuditParamsRequest.

AuditParamsRequest(email, macAddress, ipAddress, phone)
This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

Signature
AuditParamsRequest(String email, String macAddress, String ipAddress, String phone)

Parameters
email
Type: String
Email of the client that initiated the request.

macAddress
Type: String
Mac address of the customer’s device. Gateways often use this data in risk checks.

ipAddress
Type: String
The customer’s IP address. Gateways often use this data in risk checks.

phone
Type: String
Phone number of the client that initiated the request.

AuditParamsRequest Properties

The following are properties for AuditParamsRequest.

IN THIS SECTION:

email
Email of the client that initiated the request.

ipAddress
The customer’s IP address. Gateways often use this data in risk checks.
macAddress
Mac address of the customer’s device. Gateways often use this data in risk checks.

phone
Phone number of the client that initiated the request.

e-mail
Email of the client that initiated the request.

Property Value
Type: String

ipAddress
The customer’s IP address. Gateways often use this data in risk checks.

Property Value
Type: String

macAddress
Mac address of the customer’s device. Gateways often use this data in risk checks.

Property Value
Type: String

phone
Phone number of the client that initiated the request.

Property Value
Type: String

AbstractTransactionResponse Class
Abstract class for storing normalized information sent from payment gateways about a payment transaction. Holds the common response fields sent from payment gateways for authorization, sale, capture, and refund transactions.

Namespace
CommercePayments
Usage

Specify the CommercePayments namespace when creating an instance of this class. The constructor of this class takes no arguments. For example:

```java
```

IN THIS SECTION:

AbstractTransactionResponse Methods

AbstractTransactionResponse Methods

The following are methods for AbstractTransactionResponse.

**IN THIS SECTION:**

- `setAmount(amount)`: Sets the transaction amount. Must be a non-negative value.
- `setGatewayAvsCode(gatewayAvsCode)`: Sets the AVS (address verification system) result code that the gateway returned. Maximum length of 64 characters.
- `setGatewayDate(gatewayDate)`: Sets the date that the notification occurred. Some gateways don’t send this value.
- `setGatewayMessage(gatewayMessage)`: Sets error messages that the gateway returned for the notification request. Maximum length of 255 characters.
- `setGatewayReferenceDetails(gatewayReferenceDetails)`: Sets the payment gateway’s reference details.
- `setGatewayReferenceNumber(gatewayReferenceNumber)`: Sets the payment gateway’s reference number.
- `setGatewayResultCode(gatewayResultCode)`: Sets a gateway-specific result code. You can map the result code to a Salesforce-specific result code. Maximum length of 64 characters.
- `setGatewayResultCodeDescription(gatewayResultCodeDescription)`: Sets a description of the gateway-specific result code that a payment gateway returned. Maximum length of 1000 characters.
- `setSalesforceResultCodeInfo(salesforceResultCodeInfo)`: Sets the Salesforce-specific result code information.

**setAmount(amount)**

Sets the transaction amount. Must be a non-negative value.

**Signature**

```java
global void setAmount(Double amount)
```
Parameters

*amount*

Type: **Double**

The amount of the transaction.

Return Value

Type: void

**setGatewayAvsCode (gatewayAvsCode)**

Sets the AVS (address verification system) result code that the gateway returned. Maximum length of 64 characters.

Signature

```java
global void setGatewayAvsCode(String gatewayAvsCode)
```

Parameters

*gatewayAvsCode*

Type: **String**

Used to verify the address mapped to a payment method when the payments platform requests tokenization from the payment gateway.

Return Value

Type: void

**setGatewayDate (gatewayDate)**

Sets the date that the notification occurred. Some gateways don’t send this value.

Signature

```java
global void setGatewayDate(Datetime gatewayDate)
```

Parameters

*gatewayDate*

Type: **Datetime**

The date that the transaction occurred.

Return Value

Type: void

**setGatewayMessage (gatewayMessage)**

Sets error messages that the gateway returned for the notification request. Maximum length of 255 characters.
Signature

```java
global void setGatewayMessage(String gatewayMessage)
```

Parameters

`gatewayMessage`

Type: `String`

The message that the gateway returned with the transaction request. Contains additional information about the transaction.

Return Value

Type: `void`

`setGatewayReferenceDetails` (gatewayReferenceDetails)

Sets the payment gateway’s reference details.

Signature

```java
global void setGatewayReferenceDetails(String gatewayReferenceDetails)
```

Parameters

`gatewayReferenceDetails`

Type: `String`

Provides information about the gateway communication.

Return Value

Type: `void`

`setGatewayReferenceNumber` (gatewayReferenceNumber)

Sets the payment gateway’s reference number.

Signature

```java
global void setGatewayReferenceNumber(String gatewayReferenceNumber)
```

Parameters

`gatewayReferenceNumber`

Type: `String`

Unique transaction ID created by the payment gateway.

Return Value

Type: `void`
**setGatewayResultCode (gatewayResultCode)**
Sets a gateway-specific result code. You can map the result code to a Salesforce-specific result code. Maximum length of 64 characters.

**Signature**

```java
global void setGatewayResultCode(String gatewayResultCode)
```

**Parameters**

- `gatewayResultCode` Type: `String`  
  Gateway-specific result code. Must be mapped to a Salesforce-specific result code.

**Return Value**

Type: void

**setGatewayResultCodeDescription (gatewayResultCodeDescription)**
Sets a description of the gateway-specific result code that a payment gateway returned. Maximum length of 1000 characters.

**Signature**

```java
global void setGatewayResultCodeDescription(String gatewayResultCodeDescription)
```

**Parameters**

- `gatewayResultCodeDescription` Type: `String`  
  Provides additional information about the result code and why the gateway returned the specific code. Descriptions vary between different gateways.

**Return Value**

Type: void

**setSalesforceResultCodeInfo (salesforceResultCodeInfo)**
Sets the Salesforce-specific result code information.

**Signature**

```java
```

**Parameters**

- `salesforceResultCodeInfo` Type: `commercepayments.SalesforceResultCodeInfo` on page 377
Payment gateways have many response codes for payment calls. Salesforce uses the result code information to map payment gateway codes to a predefined set of standard Salesforce result codes.

Return Value
Type: void

AuthApiPaymentMethodRequest Class
Sends information about a payment method to a gateway adapter during an authorization service call.

Namespace
CommercePayments

Usage
Contains information about the payment method that is used for an authorization request. It contains all available payment methods as fields, but populates only one field for each request. The gateway adapter uses this class when constructing an authorization request. An object of this class is available through the paymentMethod field on the AuthorizationRequest Class object.

IN THIS SECTION:
  AuthApiPaymentMethodRequest Constructors
  AuthApiPaymentMethodRequest Properties

AuthApiPaymentMethodRequest Constructors
The following are constructors for AuthApiPaymentMethodRequest.

IN THIS SECTION:
  AuthApiPaymentMethodRequest(cardPaymentMethodRequest)
  AuthApiPaymentMethodRequest()

AuthApiPaymentMethodRequest(cardPaymentMethodRequest)
Constructs a sample cardPaymentMethodRequest. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

Signature
```
global AuthApiPaymentMethodRequest(commercepayments.CardPaymentMethodRequest cardPaymentMethodRequest)
```
Parameters

cardPaymentMethodRequest
Type: commercepayments.CardPaymentMethodRequest on page 315
Contains information about the card payment method. Used to send information to a gateway adapter during a service call.

AuthApiPaymentMethodRequest()
Constructor for AuthApiPaymentMethodRequest.

Signature

global AuthApiPaymentMethodRequest()

AuthApiPaymentMethodRequest Properties
The following are properties for AuthApiPaymentMethodRequest.

IN THIS SECTION:
cardPaymentMethod
The card payment method object used in a payment method request.

cardPaymentMethod
The card payment method object used in a payment method request.

Signature

global commercepayments.CardPaymentMethodRequest cardPaymentMethod {get; set;}

Property Value
Type: commercepayments.CardPaymentMethodRequest on page 315

AuthorizationReversalRequest Class
Sends information about an authorization reversal request to a gateway adapter during a service call.

Namespace
CommercePayments on page 253

Example

Add your reversal classes to your payment gateway adapter. We recommend adding AuthorizationReversal as a possible requestType value when calling processRequest on the gateway’s response.

```cpp
global commercepayments.GatewayResponse processRequest(commercepayments.paymentGatewayContext gatewayContext) {
```
commercepayments.RequestType requestType = gatewayContext.getPaymentRequestType();

commercepayments.GatewayResponse response;

try {
    //add other requestType values here
    //..
    else if (requestType == commercepayments.RequestType.AuthorizationReversal) {
        response =
            createAuthReversalResponse((commercepayments.AuthorizationReversalRequest)gatewayContext.getPaymentRequest());
    }
    return response;

Then, add a class that sets the amount of the authorization reversal request, as well as gateway information and the Salesforce result code.

global commercepayments.GatewayResponse
createAuthReversalResponse(commercepayments.AuthorizationReversalRequest authReversalRequest)
{
    if(authReversalRequest.amount!=null )
    {
        authReversalResponse.setAmount(authReversalRequest.amount);
    }
    else
    {
        throw new SalesforceValidationException('Required Field Missing : Amount');
    }
    system.debug('Response - success');
    authReversalResponse.setGatewayDate(system.now());
    authReversalResponse.setGatewayResultCode('00');
    authReversalResponse.setGatewayResultCodeDescription('Transaction Normal');
    authReversalResponse.setGatewayReferenceNumber('SF'+getRandomNumber(6));
    authReversalResponse.setSalesforceResultCodeInfo(SUCCESS_SALESFORCE_RESULT_CODE_INFO);
    return authReversalResponse;
}

IN THIS SECTION:
AuthorizationReversalRequest Constructors
AuthorizationReversalRequest Properties
AuthorizationReversalRequest Methods

AuthorizationReversalRequest Constructors
The following are constructors for AuthorizationReversalRequest.
AuthorizationReversalRequest(amount, authorizationId)

Constructor for building the amount in an authorization reversal request. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

Signature

global AuthorizationReversalRequest(Double amount, String authorizationId)

Parameters

amount
Type: Double
The amount of the authorization reversal request.

authorizationId
Type: String
The authorization request to be reversed.

AuthorizationReversalRequest Properties

The following are properties for AuthorizationReversalRequest.

IN THIS SECTION:

accountId
References the customer account for the transaction where the authorization reversal was performed.

amount
The total amount of the authorization reversal request. Can be positive or negative.

paymentAuthorizationId
References the payment authorization to be reversed.

accountId
References the customer account for the transaction where the authorization reversal was performed.

Signature

global String accountId {get; set;}

Property Value
Type: String
amount
The total amount of the authorization reversal request. Can be positive or negative.

Signature

global Double amount {get; set;}

Property Value
Type: Double

paymentAuthorizationId
References the payment authorization to be reversed.

Signature

global String paymentAuthorizationId {get; set;}

Property Value
Type: String

AuthorizationReversalRequest Methods
The following are methods for AuthorizationReversalRequest.

IN THIS SECTION:

equals(obj)
Maintains the integrity of lists of type AuthorizationReversalRequest by determining the equality of external objects in a list. This method is dynamic and based on the equals method in Java.

hashCode()
Maintains the integrity of lists of type AuthorizationReversalRequest by determining the uniqueness of the external object in a list.

toString()
Converts a date to a string.

equals(obj)
Maintains the integrity of lists of type AuthorizationReversalRequest by determining the equality of external objects in a list. This method is dynamic and based on the equals method in Java.

Signature

global Boolean equals(Object obj)
Parameters

`obj`
Type: Object
External object whose key is to be validated.

Return Value
Type: Boolean

`hashCode()`
Maintains the integrity of lists of type `AuthorizationReversalRequest` by determining the uniqueness of the external object in a list.

Signature
```
global Integer hashCode()
```

Return Value
Type: Integer

`toString()`
Converts a date to a string.

Signature
```
global String toString()
```

Return Value
Type: String

`AuthorizationReversalResponse Class`
Response sent by the payment gateway following a payment authorization reversal service.

Namespace
`CommercePayments`

Usage
The constructor of this class takes no arguments. For example:
```
```
Contains information about the payment gateway’s response following an authorization reversal transaction. The gateway adapter uses the payment gateway’s response to populate the AuthorizationReversalResponse fields. The payments platform uses the information from this class to construct the authorization gateway response shown to the user.

Example

This class builds an authorization reversal response that contains the amount of the original reversal request, gateway information, and the Salesforce result code.

```java
{
    if(authReversalRequest.amount!=null)
    {
        authReversalResponse.setAmount(authReversalRequest.amount);
    }
    else
    {
        throw new SalesforceValidationException('Required Field Missing : Amount');
    }

    system.debug('Response - success');
    authReversalResponse.setGatewayDate(system.now());
    authReversalResponse.setGatewayResultCode('00');
    authReversalResponse.setGatewayResultCodeDescription('Transaction Normal');
    authReversalResponse.setGatewayReferenceNumber('SF'+getRandomNumber(6));
    authReversalResponse.setSalesforceResultCodeInfo(SUCCESS_SALESFORCE_RESULT_CODE_INFO);
    return authReversalResponse;
}
```

IN THIS SECTION:

AuthorizationReversalResponse Methods

AuthorizationReversalResponse Methods

The following are methods for AuthorizationReversalResponse.

IN THIS SECTION:

setAmount(amount)
Contains the amount of the authorization reversal. Must be a non-zero value.

setGatewayAvsCode(gatewayAvsCode)
Sets the AVS (Address Verification System) result code that the gateway returned. Maximum length of 64 characters.

setGatewayDate(gatewayDate)
Sets the date that the authorization reversal request occurred in the payment gateway. Some gateways don’t send this value.
setGatewayMessage(gatewayMessage)
Sets error messages that the gateway returned for the authorization reversal request. Maximum length of 255 characters.

setGatewayReferenceDetails(gatewayReferenceDetails)
Stores data that you can use for subsequent authorizations. You can use any data that isn't normalized in financial entities. This field has a maximum length of 1000 characters and can store data as JSON or XML.

setGatewayReferenceNumber(gatewayReferenceNumber)
Sets a unique gateway reference number for the transaction that the gateway returned. Maximum length of 255 characters.

setGatewayResultCode(gatewayResultCode)
Sets a gateway-specific result code. The code can be mapped to a Salesforce-specific result code. Maximum length of 64 characters.

setGatewayResultCodeDescription(gatewayResultCodeDescription)
Sets a description of the gateway-specific result code that a payment gateway returned. Maximum length of 1000 characters.

setSalesforceResultCodeInfo(salesforceResultCodeInfo)
Sets the Salesforce-specific result code information. Payment gateways have many response codes for payment calls. Salesforce uses the result code information to map payment gateway codes to a predefined set of standard Salesforce result codes.

setAmount(amount)
Contains the amount of the authorization reversal. Must be a non-zero value.

**Signature**

```global void setAmount(Double amount)```

**Parameters**

- **amount**
  Type: Double

**Return Value**

Type: void

setGatewayAvsCode(gatewayAvsCode)
Sets the AVS (Address Verification System) result code that the gateway returned. Maximum length of 64 characters.

**Signature**

```global void setGatewayAvsCode(String gatewayAvsCode)```

**Parameters**

- **gatewayAvsCode**
  Type: String
    
    Used to verify the address mapped to a payment method when the payments platform requests tokenization from the payment gateway.
Return Value
Type: void

**setGatewayDate(gatewayDate)**
Sets the date that the authorization reversal request occurred in the payment gateway. Some gateways don’t send this value.

**Signature**
```
global void setGatewayDate(Datetime gatewayDate)
```

**Parameters**
- **gatewayDate**
  Type: Datetime

Return Value
Type: void

**setGatewayMessage(gatewayMessage)**
Sets error messages that the gateway returned for the authorization reversal request. Maximum length of 255 characters.

**Signature**
```
global void setGatewayMessage(String gatewayMessage)
```

**Parameters**
- **gatewayMessage**
  Type: String

Return Value
Type: void

**setGatewayReferenceDetails(gatewayReferenceDetails)**
Stores data that you can use for subsequent authorizations. You can use any data that isn’t normalized in financial entities. This field has a maximum length of 1000 characters and can store data as JSON or XML.

**Signature**
```
global void setGatewayReferenceDetails(String gatewayReferenceDetails)
```

**Parameters**
- **gatewayReferenceDetails**
  Type: String
Return Value
Type: void

**setGatewayReferenceNumber**: *(gatewayReferenceNumber)*
Sets a unique gateway reference number for the transaction that the gateway returned. Maximum length of 255 characters.

**Signature**
global void setGatewayReferenceNumber(String gatewayReferenceNumber)

**Parameters**
gatewayReferenceNumber
   - Type: String
     - Unique reference ID created by the payment gateway.

Return Value
Type: void

**setGatewayResultCode**: *(gatewayResultCode)*
Sets a gateway-specific result code. The code can be mapped to a Salesforce-specific result code. Maximum length of 64 characters.

**Signature**
global void setGatewayResultCode(String gatewayResultCode)

**Parameters**
gatewayResultCode
   - Type: String
     - Gateway-specific result code. Must be used to map a Salesforce-specific result code.

Return Value
Type: void

**setGatewayResultCodeDescription**: *(gatewayResultCodeDescription)*
Sets a description of the gateway-specific result code that a payment gateway returned. Maximum length of 1000 characters.

**Signature**
global void setGatewayResultCodeDescription(String gatewayResultCodeDescription)
Parameters

gatewayResultCodeDescription
Type: String
Description of the gateway’s result code. Use this field to learn more about why the gateway returned a certain result code.

Return Value
Type: void

setSalesforceResultCodeInfo(salesforceResultCodeInfo)
Sets the Salesforce-specific result code information. Payment gateways have many response codes for payment calls. Salesforce uses the result code information to map payment gateway codes to a predefined set of standard Salesforce result codes.

Signature


Parameters

salesforceResultCodeInfo
Type: SalesforceResultCodeInfo
Description of the Salesforce result code value.

Return Value
Type: void

AuthorizationRequest Class

Sends information about an authorization request to a gateway adapter during a service call.

Namespace

CommercePayments

Usage

This class contains information about a transaction authorization request. The gateway adapter reads fields from this class while constructing an authorization JSON request to send to the payment gateway. An object of this class is available by calling getPaymentRequest() in the PaymentGatewayContext Class.
Example

Creating a `buildAuthRequest` class to store information about the authorization request.

```java
private String buildAuthRequest(commercepayments.AuthorizationRequest authRequest) {
    // Multiply amount by 100.0 to convert to cents
    String requestBody = createRequestBody(String.ValueOf((authRequest.amount*100.0).intValue()), authRequest);
    return requestBody;
}

private String createRequestBody(String amount, commercepayments.AuthorizationRequest authRequest) {
    JSONGenerator jsonGeneratorInstance = JSON.createGenerator(true);
    String currencyIso = authRequest.currencyIsoCode;
    commercepayments.AuthApiPaymentMethodRequest paymentMethod = authRequest.paymentMethod;
    commercepayments.GatewayErrorResponse error;
    // Write data to the JSON string.
    jsonGeneratorInstance.writeStartObject();
    jsonGeneratorInstance.writeStringField('merchantAccount', '{!$Credential.Username}');
    jsonGeneratorInstance.writeStringField('reference', authRequest.comments == null ? 'randomstring' : authRequest.comments);
    if(currencyIso == null) {
        currencyIso = UserInfo.getDefaultCurrency();
    }
    jsonGeneratorInstance.writeFieldName('amount');
    jsonGeneratorInstance.writeStartObject();
    jsonGeneratorInstance.writeStringField('value', amount);
    jsonGeneratorInstance.writeStringField('currency', currencyIso);
    jsonGeneratorInstance.writeEndObject();
    commercepayments.CardPaymentMethodRequest cardPaymentMethod;
    if(paymentMethod != null) {
        cardPaymentMethod = paymentMethod.cardPaymentMethod;
        if (cardPaymentMethod != null) {
            if (cardPaymentMethod.CardCategory != null) {
                if (commercepayments.CardCategory.CreditCard == cardPaymentMethod.CardCategory) {
                    jsonGeneratorInstance.writeFieldName('card');
                    jsonGeneratorInstance.writeStartObject();
                    if (cardPaymentMethod.cvv != null)
                        jsonGeneratorInstance.writeStringField('cvc', String.ValueOf(cardPaymentMethod.cvv));
                    if (cardPaymentMethod.cardholdername != null)
                        jsonGeneratorInstance.writeStringField('holderName', cardPaymentMethod.cardholdername);
                    if (cardPaymentMethod.cardnumber != null)
                        jsonGeneratorInstance.writeStringField('number', cardPaymentMethod.cardnumber);
                    if (cardPaymentMethod.expiryMonth != null &&
                        cardPaymentMethod.expiryYear != null)
                        String expMonth =
```
((String.valueOf(cardPaymentMethod.expiryMonth)).length() == 1 ? '0' : '') +
String.valueOf(cardPaymentMethod.expiryMonth);
    jsonGeneratorInstance.writeStringField('expiryMonth', expMonth);
    jsonGeneratorInstance.writeStringField('expiryYear',
    String.valueOf(cardPaymentMethod.expiryYear));
} else {
    //Support for other card type
} else {
    throw new SampleValidationException('Required Field Missing :
CardCategory');
} else {
    throw new SampleValidationException('Required Field Missing :
CardPaymentMethod');
} else {
    throw new SampleValidationException('Required Field Missing :
PaymentMethod');
}
jsonGeneratorInstance.writeObject();
return jsonGeneratorInstanceAsString();

IN THIS SECTION:
AuthorizationRequest Constructors
AuthorizationRequest Properties
AuthorizationRequest Methods

AuthorizationRequest Constructors
The following are constructors for AuthorizationRequest.

IN THIS SECTION:
AuthorizationRequest(amount)
Constructor for building the amount in an authorization request. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

AuthorizationRequest(amount)
Constructor for building the amount in an authorization request. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

Signature

global AuthorizationRequest(Double amount)
Parameters

amount
Type: Double
The amount of the authorization.

AuthorizationRequest Properties

The following are properties for AuthorizationRequest.

IN THIS SECTION:

accountId
The customer account where the authorization is performed.
amount
The total amount of the authorization. Can be positive or negative.
comments
Comments about the authorization. Users can enter comments to provide additional information.
currencyIsoCode
The ISO currency code for the authorization request.
paymentMethod
The payment method used to process the authorization in the authorization request.

accountId
The customer account where the authorization is performed.

Signature
global String accountId {get; set;}

Property Value
Type: String

amount
The total amount of the authorization. Can be positive or negative.

Signature
global Double amount {get; set;}

Property Value
Type: Double
comments
Comments about the authorization. Users can enter comments to provide additional information.

Signature

```
global String comments {get; set;}
```

Property Value
Type: String

currencyIsOCodE
The ISO currency code for the authorization request.

Signature

```
global String currencyIsoCode {get; set;}
```

Property Value
Type: String

paymentMethod
The payment method used to process the authorization in the authorization request.

Signature

```
global AuthApiPaymentMethodRequest paymentMethod {get; set;}
```

Property Value
Type: AuthApiPaymentMethodRequest on page 270

AuthorizationRequest Methods
The following are methods for AuthorizationRequest.

IN THIS SECTION:
  equals(obj)
Maintains the integrity of lists of type AuthorizationRequest by determining the equality of external objects in a list. This method is dynamic and based on the equals method in Java.
  hashCode()
Maintains the integrity of lists of type AuthorizationRequest by determining the uniqueness of the external object in a list.
  toString()
Converts a date to a string.
equals(obj)
Maintains the integrity of lists of type AuthorizationRequest by determining the equality of external objects in a list. This method is dynamic and based on the equals method in Java.

Signature

```
global Boolean equals(Object obj)
```

Parameters

```
obj
Type: Object
External object whose key is to be validated.
```

Return Value

Type: Boolean

hashCode()
Maintains the integrity of lists of type AuthorizationRequest by determining the uniqueness of the external object in a list.

Signature

```
global Integer hashCode()
```

Return Value

Type: Integer

toString()
Converts a date to a string.

Signature

```
global String toString()
```

Return Value

Type: String

AuthorizationResponse Class
Response sent by the payment gateway adapter for an authorization service.

Namespace

CommercePayments
Usage

The constructor of this class takes no arguments. For example:

```
```

Contains information about the payment gateway’s response following an authorization transaction. The gateway adapter uses the payment gateway’s response to populate the `AuthorizationResponse` fields. The payments platform uses the information from this class to construct the authorization gateway response shown to the user.

Example

```java
private commercepayments.GatewayResponse createAuthResponse(HttpResponse response, Double amount) {
    Map<String, Object> mapOfResponseValues = (Map<String, Object>) JSON.deserializeUntyped(response.getBody());


    String resultCode = (String)mapOfResponseValues.get('resultCode');
    if(resultCode != null){
        system.debug('Response - success');
        if(resultCode.equals('Authorised')){
            system.debug('status - authorised');
            authResponse.setGatewayAuthCode((String)mapOfResponseValues.get('authCode'));
        } else {
            //Sample returns 200 with refused status in some cases
            system.debug('status - refused');
            authResponse.setGatewayResultCodeDescription((String)mapOfResponseValues.get('refusalReason'));
        }
    } else {
        system.debug('Response - failed');
        string statusCode = (String)mapOfResponseValues.get('errorType');
        string message = (String)mapOfResponseValues.get('message');
        commercepayments.GatewayErrorResponse error = new commercepayments.GatewayErrorResponse(errorType, message);
    }

    authResponse.setGatewayReferenceNumber((String)mapOfResponseValues.get('pspReference'));
    authResponse.setAmount(amount);
    authResponse.setGatewayDate(system.now());
    return authResponse;
}
```
IN THIS SECTION:
AuthorizationResponse Methods

AuthorizationResponse Methods
The following are methods for AuthorizationResponse.

setAmount(amount)
Sets the amount of the authorization. Must be a non-zero value.

setAuthorizationExpirationDate(authExpDate)
Sets the expiration date of the authorization request.

setGatewayAuthCode(gatewayAuthCode)
Sets the authorization code that the gateway returned. Maximum length of 64 characters.

setGatewayAvsCode(gatewayAvsCode)
Sets the AVS (address verification system) result code information that the gateway returned. Maximum length of 64 characters.

setGatewayDate(gatewayDate)
Sets the date that the authorization occurred. Some gateways don’t send this value.

setGatewayMessage(gatewayMessage)
Sets error messages that the gateway returned for the authorization request. Maximum length of 255 characters.

setGatewayReferenceDetails(gatewayReferenceDetails)
Stores data that you can use for subsequent authorizations. You can use any data that isn’t normalized in financial entities. This field has a maximum length of 1000 characters and can store data as JSON or XML.

setGatewayReferenceNumber(gatewayReferenceNumber)
Sets the unique gateway reference number for the transaction that the gateway returned. Maximum length of 255 characters.

setGatewayResultCode(gatewayResultCode)
Sets a gateway-specific result code. The code can be mapped to a Salesforce-specific result code. Maximum length of 64 characters.

setGatewayResultCodeDescription(gatewayResultCodeDescription)
Sets a description of the gateway-specific result code that a payment gateway returned. Maximum length of 1000 characters.

setPaymentMethodTokenizationResponse(paymentMethodTokenizationResponse)
Sets information from the gateway about the tokenized payment method.

setSalesforceResultCodeInfo(salesforceResultCodeInfo)
Sets the Salesforce-specific result code information. Payment gateways have many response codes for payment calls. Salesforce uses the result code information to map payment gateway codes to a predefined set of standard Salesforce result codes.
setAmount(amount)
Sets the amount of the authorization. Must be a non-zero value.

Signature
global void setAmount(Double amount)

Parameters
amount
Type: Double

Return Value
Type: void

setAuthorizationExpirationDate(authExpDate)
Sets the expiration date of the authorization request.

Signature
global void setAuthorizationExpirationDate(Datetime authExpDate)

Parameters
authExpDate
Type: Datetime

Return Value
Type: void

setGatewayAuthCode(gatewayAuthCode)
Sets the authorization code that the gateway returned. Maximum length of 64 characters.

Signature
global void setGatewayAuthCode(String gatewayAuthCode)

Parameters
gatewayAuthCode
Type: String
The authorization code returned by the gateway.

Return Value
Type: void
**setGatewayAvsCode (gatewayAvsCode)**
Sets the AVS (address verification system) result code information that the gateway returned. Maximum length of 64 characters.

**Signature**
`global void setGatewayAvsCode(String gatewayAvsCode)`

**Parameters**
gatewayAvsCode  
Type: `String`  
Used to verify the address mapped to a payment method when the payments platform requests tokenization from the payment gateway.

**Return Value**
Type: void

**setGatewayDate (gatewayDate)**
Sets the date that the authorization occurred. Some gateways don’t send this value.

**Signature**
`global void setGatewayDate(Datetime gatewayDate)`

**Parameters**
gatewayDate  
Type: `Datetime`  

**Return Value**
Type: void

**setGatewayMessage (gatewayMessage)**
Sets error messages that the gateway returned for the authorization request. Maximum length of 255 characters.

**Signature**
`global void setGatewayMessage(String gatewayMessage)`

**Parameters**
gatewayMessage  
Type: `String`
Return Value
Type: void

**setGatewayReferenceDetails** (gatewayReferenceDetails)
Stores data that you can use for subsequent authorizations. You can use any data that isn’t normalized in financial entities. This field has a maximum length of 1000 characters and can store data as JSON or XML.

Signature
```
global void setGatewayReferenceDetails(String gatewayReferenceDetails)
```

Parameters
`gatewayReferenceDetails`
Type: String

Return Value
Type: void

**setGatewayReferenceNumber** (gatewayReferenceNumber)
Sets the unique gateway reference number for the transaction that the gateway returned. Maximum length of 255 characters.

Signature
```
global void setGatewayReferenceNumber(String gatewayReferenceNumber)
```

Parameters
`gatewayReferenceNumber`
Type: String

Unique authorization ID created by the payment gateway.

Return Value
Type: void

**setGatewayResultCode** (gatewayResultCode)
Sets a gateway-specific result code. The code can be mapped to a Salesforce-specific result code. Maximum length of 64 characters.

Signature
```
global void setGatewayResultCode(String gatewayResultCode)
```
Parameters

gatewayResultCode
Type: String

Gateway-specific result code. Must be used to map a Salesforce-specific result code.

Return Value
Type: void

setGatewayResultCodeDescription(gatewayResultCodeDescription)

Sets a description of the gateway-specific result code that a payment gateway returned. Maximum length of 1000 characters.

Signature

global void setGatewayResultCodeDescription(String gatewayResultCodeDescription)

Parameters

gatewayResultCodeDescription
Type: String

Description of the gateway's result code. Use this field to learn more about why the gateway returned a certain result code.

Return Value
Type: void

setPaymentMethodTokenizationResponse(paymentMethodTokenizationResponse)

Sets information from the gateway about the tokenized payment method.

Signature

global void setPaymentMethodTokenizationResponse(commercepayments.PaymentMethodTokenizationResponse paymentMethodTokenizationResponse)

Parameters

paymentMethodTokenizationResponse
PaymentMethodTokenizationResponse on page 342

Gateway response sent by payment gateway adapters for the payment method tokenization request.

Return Value
Type: void
**setSalesforceResultCodeInfo(salesforceResultCodeInfo)**

Sets the Salesforce-specific result code information. Payment gateways have many response codes for payment calls. Salesforce uses the result code information to map payment gateway codes to a predefined set of standard Salesforce result codes.

**Signature**

```java
```

**Parameters**

- `salesforceResultCodeInfo`  
  - Type: `SalesforceResultCodeInfo` on page 377  
  - Description of the Salesforce result code value.

**Return Value**

Type: `void`

---

**BaseApiPaymentMethodRequest Class**

Abstract class used to send information about a payment method to a gateway adapter during a service call.

**Namespace**

`CommercePayments`

**Usage**

`BaseApiPaymentMethodRequest` is the base class for `SaleApiPaymentMethodRequest` and `AuthApiPaymentMethodRequest`.

**BaseApiPaymentMethodRequest Constructors**

The following are constructors for `BaseApiPaymentMethodRequest`.

**BaseApiPaymentMethodRequest(address, id, saveForFuture)**

Constructs a payment method. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.
**BaseApiPaymentMethodRequest(address, id, saveForFuture)**

Constructs a payment method. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

**Signature**

```java
global BaseApiPaymentMethodRequest(commercepayments.AddressRequest address, String id, Boolean saveForFuture)
```

**Parameters**

- **address**
  - Type: `commercepayments.AddressRequest` on page 259
  - Sends data related on address request to a gateway adapter during a service call.
- **id**
  - Type: `String`
- **saveForFuture**
  - Type: `Boolean`
  - Indicates whether Salesforce saves the payment method for future use.

**BaseApiPaymentMethodRequest Properties**

The following are properties for `BaseApiPaymentMethodRequest`.

**IN THIS SECTION:**

- **address**
  - The payment method’s address.
- **id**
  - ID of the payment method request.
- **saveForFuture**
  - Indicates whether the payment method is saved as a record in Salesforce for future use.

**address**

The payment method’s address.

**Signature**

```java
global commercepayments.AddressRequest address {get; set;}
```

**Property Value**

- Type: `AddressRequest` on page 259
**id**
ID of the payment method request.

**Signature**
global String id {get; set;}

**Property Value**
Type: String

**saveForFuture**
Indicates whether the payment method is saved as a record in Salesforce for future use.

**Signature**
global Boolean saveForFuture {get; set;}

**Property Value**
Type: Boolean

**BaseApiPaymentMethodRequest Methods**
The following are methods for `BaseApiPaymentMethodRequest`.

IN THIS SECTION:
- equals(obj)
  Maintains the integrity of lists of type `BaseApiPaymentMethodRequest` by determining the equality of external objects in a list. This method is dynamic and is based on the equals method in Java.
- hashCode()
  Maintains the integrity of lists of type `BaseApiPaymentMethodRequest` by determining the uniqueness of the external object records in a list.
- toString()
  Converts a date to a string.

**equals(obj)**
Maintains the integrity of lists of type `BaseApiPaymentMethodRequest` by determining the equality of external objects in a list. This method is dynamic and is based on the equals method in Java.

**Signature**
global Boolean equals(Object obj)
Parameters

\( obj \)
Type: Object
External object whose key is to be validated.

Return Value
Type: Boolean

**hashCode()**
Maintains the integrity of lists of type `BaseApiPaymentMethodRequest` by determining the uniqueness of the external object records in a list.

Signature

```
global Integer hashCode()
```

Return Value
Type: Integer

**toString()**
Converts a date to a string.

Signature

```
global String toString()
```

Return Value
Type: String

**BaseNotification Class**
Abstract class for storing notification information sent from payment gateways.

**Namespace**
`CommercePayments`

**Usage**
An abstract class that contains the common fields from payment gateways. `BaseNotification` can't be instantiated on its own. The constructor of this class takes no arguments. For example:

```
CommercePayments.BaseNotification bnt = new CommercePayments.BaseNotification();
```
Example

```java
commercepayments.BaseNotification notification = null;
if ('CAPTURE'.equals(eventCode)) {
    notification = new commercepayments.CaptureNotification();
} else if ('REFUND'.equals(eventCode)) {
    notification = new commercepayments.ReferencedRefundNotification();
}
```

IN THIS SECTION:

**BaseNotification Methods**

The following are methods for `BaseNotification`.

**setAmount(amount)**
Sets the transaction amount. Must be a non-negative value.

**setGatewayDate(gatewayDate)**
Sets the date that the notification occurred. Some gateways don’t send this value.

**setGatewayMessage(gatewayMessage)**
Sets error messages that the gateway returned for the notification request. Maximum length of 255 characters.

**setGatewayReferenceDetails(gatewayReferenceDetails)**
Sets the payment gateway’s reference details.

**setGatewayReferenceNumber(gatewayReferenceNumber)**
Sets the payment gateway’s reference number.

**setGatewayResultCode(gatewayResultCode)**
Sets a gateway-specific result code. The code can be mapped to a Salesforce-specific result code. Maximum length of 64 characters.

**setGatewayResultCodeDescription(gatewayResultCodeDescription)**
Sets a description of the gateway-specific result code that a payment gateway returned. Maximum length of 1000 characters.

**setId(id)**
Sets the ID of the notification sent by the gateway.

**setSalesforceResultCodeInfo(salesforceResultCodeInfo)**
Sets the information about the Salesforce-specific result code used to match a result code from a payment gateway.

**setStatus(status)**
Sets the status of the notification sent by the gateway.

**setAmount(amount)**
Sets the transaction amount. Must be a non-negative value.
Signature

```plaintext
global void setAmount(Double amount)
```

Parameters

`amount`  
Type: `Double`  
The amount of the transaction.

Return Value

Type: `void`

---

**setGatewayDate (gatewayDate)**

Sets the date that the notification occurred. Some gateways don’t send this value.

Signature

```plaintext
global void setGatewayDate(Datetime gatewayDate)
```

Parameters

`gatewayDate`  
Type: `Datetime`  
The date that the notification occurred.

Return Value

Type: `void`

---

**setGatewayMessage (gatewayMessage)**

Sets error messages that the gateway returned for the notification request. Maximum length of 255 characters.

Signature

```plaintext
global void setGatewayMessage(String gatewayMessage)
```

Parameters

`gatewayMessage`  
Type: `String`  
The message that the gateway returned with the notification request. Contains additional information about the notification.

Return Value

Type: `void`
setGatewayReferenceDetails(gatewayReferenceDetails)
Sets the payment gateway's reference details.

Signature

global void setGatewayReferenceDetails(String gatewayReferenceDetails)

Parameters

gatewayReferenceDetails
Type: String
Provides information about the gateway communication.

Return Value
Type: void

setGatewayReferenceNumber(gatewayReferenceNumber)
Sets the payment gateway's reference number.

Signature

global void setGatewayReferenceNumber(String gatewayReferenceNumber)

Parameters

gatewayReferenceNumber
Type: String
Unique transaction ID created by the payment gateway.

Return Value
Type: void

setGatewayResultCode(gatewayResultCode)
Sets a gateway-specific result code. The code can be mapped to a Salesforce-specific result code. Maximum length of 64 characters.

Signature

global void setGatewayResultCode(String gatewayResultCode)

Parameters

gatewayResultCode
Type: String
Gateway-specific result code. Must be used to map a Salesforce-specific result code.
Return Value
Type: void

`setGatewayResultCodeDescription(gatewayResultCodeDescription)`
Sets a description of the gateway-specific result code that a payment gateway returned. Maximum length of 1000 characters.

Signature
`global void setGatewayResultCodeDescription(String gatewayResultCodeDescription)`

Parameters
`gatewayResultCodeDescription`
Type: `String`
Provides additional information about the result code and why the gateway returned the code. Descriptions vary between different gateways.

Return Value
Type: void

`setId(id)`
Sets the ID of the notification sent by the gateway.

Signature
`global void setId(String id)`

Parameters
`id`
Type: `String`

Return Value
Type: void

`setSalesforceResultCodeInfo(salesforceResultCodeInfo)`
Sets the information about the Salesforce-specific result code used to match a result code from a payment gateway.

Signature
Parameters

\textit{salesforceResultCodeInfo}

\textbf{Type:} \texttt{commercepayments.SalesforceResultCodeInfo} on page 377

Payment gateways have many response codes for payment calls. Salesforce uses the result code information to map payment gateway codes to a predefined set of standard Salesforce result codes.

Return Value

\textbf{Type:} void

\textbf{setStatus (status)}

Sets the status of the notification sent by the gateway.

Signature

\texttt{global void setStatus(commercepayments.NotificationStatus status)}

Parameters

\textit{status}

\textbf{Type:} \texttt{commercepayments.NotificationStatus} on page 330

Shows whether the payments platform successfully received the notification from the gateway.

Return Value

\textbf{Type:} void

\textbf{BasePaymentMethodRequest Class}

Abstract class for storing information about payment methods.

Namespace

\texttt{CommercePayments}

Usage

The \texttt{BasePaymentMethodRequest} class contains fields common to \texttt{CardPaymentMethodRequest} on page 315.

IN THIS SECTION:

BasePaymentMethodRequest Methods

\textbf{BasePaymentMethodRequest Methods}

The following are methods for \texttt{BasePaymentMethodRequest}. 

300
equals(obj)
Maintains the integrity of lists of type BasePaymentMethodRequest by determining the equality of external objects in a list. This method is dynamic and based on the equals method in Java.

hashCode()
Maintains the integrity of lists of type BasePaymentMethodRequest by determining the uniqueness of the external object records in a list.

toString()
Converts a date to a string.

equals(obj)
Maintains the integrity of lists of type BasePaymentMethodRequest by determining the equality of external objects in a list. This method is dynamic and based on the equals method in Java.

Signature

global Boolean equals(Object obj)

Parameters

obj
Type: Object
External object whose key is to be validated.

Return Value
Type: Boolean

hashCode()
Maintains the integrity of lists of type BasePaymentMethodRequest by determining the uniqueness of the external object records in a list.

Signature

global Integer hashCode()

Return Value
Type: Integer

toString()
Converts a date to a string.

Signature

global String toString()
Return Value
Type: String

BaseRequest Class
BaseRequest is extended by all the request classes.

Namespace
CommercePayments

IN THIS SECTION:
BaseRequest Methods

BaseRequest Methods
The following are methods for BaseRequest.

IN THIS SECTION:
BaseRequest(AdditionalData, IdempotencyKey)
Used for testing.

BaseRequest(AdditionalData, IdempotencyKey)
Used for testing.

Signature
global Void BaseRequest(String AdditionalData, Map<String, String> IdempotencyKey)

Parameters
AdditionalData
Type: String
Contains additional data that may be required for a payment request. The additionalData object consists of key-value pairs. You can retrieve the additionalData object from the request object: Map<String, String> additionalData=request.additionalData

IdempotencyKey
Type: Map<String, String>
Unique value that's generated by a client and sent to the server in the request. The server stores the value and uses the it to keep track of the request status.

Return Value
Type: Void
CaptureNotification Class

When a payment gateway sends a notification for a capture transaction, the payment gateway adapter creates the `CaptureNotification` object to store information about the notification.

Namespace

`CommercePayments`

Usage

`CaptureNotification` is used in asynchronous payment gateway adapters. Specify the `CommercePayments` namespace when creating an instance of this class. The constructor of this class takes no arguments. For example:

```
CommercePayments.CaptureNotification crn = new CommercePayments.CaptureNotification();
```

Example

```
commercepayments.BaseNotification notification = null;
if ('CAPTURE'.equals(eventCode)) {
    notification = new commercepayments.CaptureNotification();
} else if ('REFUND'.equals(eventCode)) {
    notification = new commercepayments.ReferencedRefundNotification();
}
```

IN THIS SECTION:

CaptureNotification Methods

The following are methods for `CaptureNotification`.

IN THIS SECTION:

- `setAmount(amount)`
  Sets the transaction amount. Must be a non-negative value.
- `setGatewayDate(gatewayDate)`
  Sets the date that the transaction occurred. Some gateways don't send this value.
- `setGatewayMessage(gatewayMessage)`
  Sets error messages that the gateway returned for the payment request. Maximum length of 255 characters.
- `setGatewayReferenceDetails(gatewayReferenceDetails)`
  Sets additional data that you can use for subsequent transactions. You can use any data that isn't normalized in financial entities. This field has a maximum length of 1000 characters and can store data as JSON or XML.
- `setGatewayReferenceNumber(gatewayReferenceNumber)`
  Sets the unique gateway reference number for the transaction that the gateway returned. Maximum length of 255 characters.
setGatewayResultCode(gatewayResultCode)
Sets a gateway-specific result code. The code can be mapped to a Salesforce-specific result code. Maximum length of 64 characters.

setGatewayResultCodeDescription(gatewayResultCodeDescription)
Sets a description of the gateway-specific result code that a gateway returned. Maximum length of 1000 characters.

setId(id)
Sets the ID of a notification sent by the payment gateway.

setSalesforceResultCodeInfo(salesforceResultCodeInfo)
Sets the Salesforce-specific result code information. Payment gateways have many response codes for payment calls. Salesforce uses the result code information to map payment gateway codes to a predefined set of standard Salesforce result codes.

setStatus(status)
Sets the notification status to the same value that was sent by the gateway.

setAmount(amount)
Sets the transaction amount. Must be a non-negative value.

Signature

global void setAmount(Double amount)

Parameters

amount
Type: Double
The amount to be debited or captured.

Return Value

Type: void

setGatewayDate(gatewayDate)
Sets the date that the transaction occurred. Some gateways don’t send this value.

Signature

global void setGatewayDate(Datetime gatewayDate)

Parameters

gatewayDate
Type: Datetime
Date and time of the gateway communication.

Return Value

Type: void
**setGatewayMessage(gatewayMessage)**
Sets error messages that the gateway returned for the payment request. Maximum length of 255 characters.

**Signature**

```java
global void setGatewayMessage(String gatewayMessage)
```

**Parameters**

gatewayMessage
Type: String
Information on error messages sent from the gateway.

**Return Value**

Type: void

**setGatewayReferenceDetails(gatewayReferenceDetails)**
Sets additional data that you can use for subsequent transactions. You can use any data that isn't normalized in financial entities. This field has a maximum length of 1000 characters and can store data as JSON or XML.

**Signature**

```java
global void setGatewayReferenceDetails(String gatewayReferenceDetails)
```

**Parameters**

gatewayReferenceDetails
Type: String

**Return Value**

Type: void

**setGatewayReferenceNumber(gatewayReferenceNumber)**
Sets the unique gateway reference number for the transaction that the gateway returned. Maximum length of 255 characters.

**Signature**

```java
global void setGatewayReferenceNumber(String gatewayReferenceNumber)
```

**Parameters**

gatewayReferenceNumber
Type: String
Unique transaction ID created by the payment gateway.
Return Value
Type: void

**setGatewayResultCode (gatewayResultCode)**
Sets a gateway-specific result code. The code can be mapped to a Salesforce-specific result code. Maximum length of 64 characters.

Signature
```
global void setGatewayResultCode(String gatewayResultCode)
```

Parameters
```
gatewayResultCode
Type: String
Gateway-specific result code. Map this value to a Salesforce-specific result code.
```

Return Value
Type: void

**setGatewayResultCodeDescription (gatewayResultCodeDescription)**
Sets a description of the gateway-specific result code that a gateway returned. Maximum length of 1000 characters.

Signature
```
global void setGatewayResultCodeDescription(String gatewayResultCodeDescription)
```

Parameters
```
gatewayResultCodeDescription
Type: String
Description of the gateway’s result code. Use this field to learn more about why the gateway returned a certain result code.
```

Return Value
Type: void

**setId (id)**
Sets the ID of a notification sent by the payment gateway.

Signature
```
global void setId(String id)
```
Parameters

$id$
Type: String

Return Value
Type: void

`setSalesforceResultCodeInfo(salesforceResultCodeInfo)`
Sets the Salesforce-specific result code information. Payment gateways have many response codes for payment calls. Salesforce uses the result code information to map payment gateway codes to a predefined set of standard Salesforce result codes.

Signature

Parameters

`salesforceResultCodeInfo`
Type: `commercepayments.SalesforceResultCodeInfo` on page 377
Description of the Salesforce result code value.

Return Value
Type: void

`setStatus(status)`
Sets the notification status to the same value that was sent by the gateway.

Signature
`global void setStatus(commercepayments.NotificationStatus status)`

Parameters

`status`
Type: `NotificationStatus` on page 330
Sets the Salesforce-specific result code information. Payment gateways have many response codes for payment calls. Salesforce uses the result code information to map payment gateway codes to a predefined set of standard Salesforce result codes.

Return Value
Type: void
CaptureRequest Class

Represents a capture request. This class extends the BaseRequest class and inherits all its methods.

Namespace

CommercePayments on page 253

Usage

The CaptureRequest class’s buildCaptureRequest method creates a CaptureRequest object to store payment information, such as value and currency, as JSON strings.

Example

Builds a CaptureRequest object for a multicurrency org.

```java
private String buildCaptureRequest(commercepayments.CaptureRequest captureRequest) {
    Boolean IS_MULTICURRENCY_ORG = UserInfo.isMultiCurrencyOrganization();
    QueryUtils qBuilderForAuth = new QueryUtils(PaymentAuthorization.SObjectType);
    // Add required fields
    qBuilderForAuth.getSelectClause().addField('GatewayRefNumber', false);
    if (IS_MULTICURRENCY_ORG) {
        // addField also takes a boolean to enable translation (uses label instead of actual value)
        qBuilderForAuth.getSelectClause().addField('CurrencyIsoCode', false);
    }
}
```

IN THIS SECTION:

CaptureRequest Constructors

CaptureRequest Properties

CaptureRequest Constructors

The following are constructors for CaptureRequest.

IN THIS SECTION:

CaptureRequest(amount, authorizationId)

This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

CaptureRequest(amount, authorizationId)

This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

Parameters

amount

Type: Double
The amount to be debited or captured.

authorizationId

Type: String

Represents a payment authorization record.

CaptureRequest Properties

The following are properties for CaptureRequest.

IN THIS SECTION:

accountId
Account ID value. References an account record.

amount
Amount of currency that needs to be captured.

paymentAuthorizationId
ID value that references a PaymentAuthorization.

accountId

Account ID value. References an account record.

Property Value
Type: String

amount

Amount of currency that needs to be captured.

Property Value
Type: Double

paymentAuthorizationId

ID value that references a PaymentAuthorization.

Property Value
Type: String

CaptureResponse Class

The payment gateway adapter sends this response for the capture request type. This class extends AbstractResponse and inherits its methods.
Namespace

CommercePayments on page 253

Usage

You must specify the CommercePayments namespace when creating an instance of this class. The constructor of this class takes no arguments. For example:

```java
CommercePayments.CaptureResponse ctr = new CommercePayments.CaptureResponse();
```

IN THIS SECTION:

CaptureResponse Methods

The following are methods for CaptureResponse.

IN THIS SECTION:

- `setAmount(amount)`
  Sets the transaction amount.

- `setAsync(async)`
  Indicates whether the payment gateway adapter used in the payment capture was asynchronous (`true`) or synchronous (`false`).

- `setGatewayAvsCode(gatewayAvsCode)`
  Sets the payment gateway’s AVS (address verification system) code.

- `setGatewayDate(gatewayDate)`
  Sets the payment gateway’s date.

- `setGatewayMessage(gatewayMessage)`
  Sets information or messages that the gateway returned.

- `setGatewayReferenceDetails(gatewayReferenceDetails)`
  Sets the payment gateway’s reference details.

- `setGatewayReferenceNumber(gatewayReferenceNumber)`
  Sets the payment gateway’s reference number.

- `setGatewayResultCode(gatewayResultCode)`
  Sets the payment gateway’s result code.

- `setGatewayResultCodeDescription(gatewayResultCodeDescription)`
  Sets the payment gateway’s result code description.

- `setSalesforceResultCodeInfo(salesforceResultCodeInfo)`
  Sets Salesforce result code information.

### setAmount(amount)

Sets the transaction amount.
**Signature**

global void setAmount(Double amount)

**Parameters**

amount
Type: Double
The amount to be debited or captured.

**Return Value**
Type: void

**setAsync(async)**
Indicates whether the payment gateway adapter used in the payment capture was asynchronous (True) or synchronous (False).

**Signature**

global void setAsync(Boolean async)

**Parameters**

async
Type: Boolean

**Return Value**
Type: void

**setGatewayAvsCode(gatewayAvsCode)**
Sets the payment gateway's AVS (address verification system) code.

**Signature**

global void setGatewayAvsCode(String gatewayAvsCode)

**Parameters**

gatewayAvsCode
Type: String
Payment gateways that use an AVS system return this code.

**Return Value**
Type: void
setGatewayDate (gatewayDate)
Sets the payment gateway’s date.

Signature

global void setGatewayDate(Datetime gatewayDate)

Parameters

gatewayDate
  Type: Datetime
  The date that communication happened with the gateway.

Return Value

Type: void

setGatewayMessage (gatewayMessage)
Sets information or messages that the gateway returned.

Signature

global void setGatewayMessage(String gatewayMessage)

Parameters

gatewayMessage
  Type: String
  Information or error messages returned by the gateway.

Return Value

Type: void

setGatewayReferenceDetails (gatewayReferenceDetails)
Sets the payment gateway’s reference details.

Signature

global void setGatewayReferenceDetails(String gatewayReferenceDetails)

Parameters

gatewayReferenceDetails
  Type: String
  Provides information about the gateway communication.
Return Value
Type: void

`setGatewayReferenceNumber(gatewayReferenceNumber)`
Sets the payment gateway’s reference number.

Signature
`global void setGatewayReferenceNumber(String gatewayReferenceNumber)`

Parameters
`gatewayReferenceNumber`  
Type: String  
Unique transaction ID created by the payment gateway.

Return Value
Type: void

`setGatewayResultCode(gatewayResultCode)`
Sets the payment gateway’s result code.

Signature
`global void setGatewayResultCode(String gatewayResultCode)`

Parameters
`gatewayResultCode`  
Type: String  
The gateway result code. You must map this to a Salesforce result code.

Return Value
Type: void

`setGatewayResultCodeDescription(gatewayResultCodeDescription)`
Sets the payment gateway’s result code description.

Signature
`global void setGatewayResultCodeDescription(String gatewayResultCodeDescription)`
Parameters

gatewayResultCodeDescription
    Type: String
    Description of the GatewayResultCode. Provides additional context about the result code that the gateway returned.

Return Value

Type: void

setSalesforceResultCodeInfo(salesforceResultCodeInfo)

Sets Salesforce result code information.

Signature


Parameters

salesforceResultCodeInfo
    SalesforceResultCodeInfoType: commercepayments.SalesforceResultCodeInfo
    Description of the Salesforce result code value.

Return Value

Type: void

CardCategory Enum

Defines whether the payment method represents a credit card or a debit card.

Namespace

CommercePayments on page 253

Enum Values

The following are the values of the commercepayments.CardCategory enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CreditCard</td>
<td>Shows that the payment method is a credit card.</td>
</tr>
<tr>
<td>DebitCard</td>
<td>Shows that the payment method is a debit card.</td>
</tr>
</tbody>
</table>
CardPaymentMethodRequest Class

Sends data related to a card payment method to a gateway adapter during a service call.

Namespace

CommercePayments on page 253

Usage

This class contains details about the card used as a payment method for authorization, sale, or tokenization transaction requests. The gateway adapter reads the fields of this class object while constructing a transaction JSON request to send to the payment gateway. The object of this class is available as the cardPaymentMethod field in the SaleApiPaymentMethodRequest Class, AuthApiPaymentMethodRequest Class, and PaymentMethodTokenizationRequest Class.

Example: This code sample retrieves the cardPaymentMethod object from the paymentMethod class.

```java
commercepayments.CardPaymentMethodRequest cardPaymentMethod = paymentMethod.cardPaymentMethod;
```

IN THIS SECTION:

CardPaymentMethodRequest Constructors

CardPaymentMethodRequest Properties

CardPaymentMethodRequest Methods

CardPaymentMethodRequest Constructors

The following are constructors for CardPaymentMethodRequest.

IN THIS SECTION:

CardPaymentMethodRequest(cardCategory)

Sets the cardCategory value for the card payment method request.

CardPaymentMethodRequest<CardCategory>

Sets the cardCategory value for the card payment method request.

Signature

```
global CardPaymentMethodRequest(commercepayments.CardCategory cardCategory)
```

Parameters

cardCategory

Type: CardCategory on page 314

Defines whether the card payment method is a credit card or a debit card.
CardPaymentMethodRequest Properties

The following are properties for CardPaymentMethodRequest.

IN THIS SECTION:

  accountId
  Customer account for this payment method.

  autoPay
  Indicates whether a token is being requested so that the payment method can be used for recurring payments.

  cardCategory
  Indicates whether a card payment method is for a credit card or debit card.

  cardHolderFirstName
  The first name of the cardholder for the card payment method.

  cardHolderLastName
  The last name of the cardholder for the card payment method.

  cardHolderName
  Full name of the cardholder on the card payment method.

  cardNumber
  System-defined unique ID for the card payment method.

  cardType
  Defines the credit card bank. Possible values are AmericanExpress, DinersClub, JCB, Maestro, MasterCard, and Visa.

  cvv
  The card security code for the credit or debit card on a card payment method.

  email
  Email address of the cardholder for the credit or debit card on a card payment method.

  expiryMonth
  Expiration month for the credit or debit card on a card payment method.

  expiryYear
  Expiration year of the credit or debit card for the card payment method.

  inputCardType
  Input field for the card type. This field doesn’t store the card type directly, but instead populates CardBin, LastFour, and DisplayCardNumber based on the value entered in inputCardType.

  startMonth
  The credit or debit card becomes valid on the first day of the startMonth in the startYear.

  startYear
  Year during which the credit or debit card becomes valid.

accountId
Customer account for this payment method.
**Signature**

```csharp
global String accountId {get; set;}
```

**Property Value**

Type: `String`

---

**autoPay**

Indicates whether a token is being requested so that the payment method can be used for recurring payments.

**Signature**

```csharp
global Boolean autoPay {get; set;}
```

**Property Value**

Type: `Boolean`

---

**cardCategory**

Indicates whether a card payment method is for a credit card or debit card.

**Signature**

```csharp
global commercepayments.CardCategory cardCategory {get; set;}
```

**Property Value**

Type: `CardCategory` on page 314

---

**cardHolderFirstName**

The first name of the cardholder for the card payment method.

**Signature**

```csharp
global String cardHolderFirstName {get; set;}
```

**Property Value**

Type: `String`

---

**cardHolderLastName**

The last name of the cardholder for the card payment method.

**Signature**

```csharp
global String cardHolderLastName {get; set;}
```
Property Value
Type: String

**cardHolderName**
Full name of the cardholder on the card payment method.

**Signature**
global String cardHolderName {get; set;}

Property Value
Type: String

**cardNumber**
System-defined unique ID for the card payment method.

**Signature**
global String cardNumber {get; set;}

Property Value
Type: String

**cardType**
Defines the credit card bank. Possible values are AmericanExpress, DinersClub, JCB, Maestro, MasterCard, and Visa.

**Signature**
global commercepayments.CardType cardType {get; set;}

Property Value
Type: CardType

**cvv**
The card security code for the credit or debit card on a card payment method.

**Signature**
global String cvv {get; set;}

Property Value
Type: String
email
Email address of the cardholder for the credit or debit card on a card payment method.

Signature
global String email {get; set;}

Property Value
Type: String

expiryMonth
Expiration month for the credit or debit card on a card payment method.

Signature
global Integer expiryMonth {get; set;}

Property Value
Type: Integer

expiryYear
Expiration year of the credit or debit card for the card payment method.

Signature
global Integer expiryYear {get; set;}

Property Value
Type: Integer

inputCardType
Input field for the card type. This field doesn’t store the card type directly, but instead populates CardBin, LastFour, and DisplayCardNumber based on the value entered in inputCardType.

Signature
global String inputCardType {get; set;}

Property Value
Type: String
**startMonth**

The credit or debit card becomes valid on the first day of the `startMonth` in the `startYear`.

**Signature**

global Integer startMonth {get; set;}

**Property Value**

Type: Integer

**startYear**

Year during which the credit or debit card becomes valid.

**Signature**

global Integer startYear {get; set;}

**Property Value**

Type: Integer

**CardPaymentMethodRequest Methods**

The following are methods for `CardPaymentMethodRequest`.

**IN THIS SECTION:**

- `equals(obj)`

  Maintains the integrity of lists of type `CardPaymentMethodRequest` by determining the equality of external objects in a list. This method is dynamic and based on the equals method in Java.

- `hashCode()`

  Maintains the integrity of lists of type `CardPaymentMethodRequest`.

- `toString()`

  Converts a date to a string.

**equals(obj)**

Maintains the integrity of lists of type `CardPaymentMethodRequest` by determining the equality of external objects in a list. This method is dynamic and based on the equals method in Java.

**Signature**

global Boolean equals(Object obj)
Parameters

\textit{obj}

Type: Object

External object whose key is to be validated.

Return Value

Type: Boolean

\textbf{hashCode()}

Maintains the integrity of lists of type \texttt{CardPaymentMethodRequest}.

Signature

\texttt{global Integer hashCode()}

Return Value

Type: Integer

\textbf{toString()}

Converts a date to a string.

Signature

\texttt{global String toString()}

Return Value

Type: String

\textbf{CustomMetadataTypeInfo Class}

Access information about custom metadata. The \texttt{PaymentGatewayAdapter} can send \texttt{CustomMetadataTypeInfo} to transaction requests through the response object's \texttt{SalesforceResultCodeInfo}.

Namespace

\textit{CommercePayments} on page 253

IN THIS SECTION:

- \texttt{CustomMetadataTypeInfo Constructors}
- \texttt{CustomMetadataTypeInfo Methods}
CustomMetadataTypeInfo Constructors

The following are constructors for CustomMetadataTypeInfo.

IN THIS SECTION:

    CustomMetadataTypeInfo(cmtRecordId, cmtSfResultCodeFieldName)
    Constructor for providing custom metadata type information.

CustomMetadataTypeInfo (cmtRecordId, cmtSfResultCodeFieldName)

Constructor for providing custom metadata type information.

Signature

global CustomMetadataTypeInfo(String cmtRecordId, String cmtSfResultCodeFieldName)

Parameters

cmtRecordId
    Type: String
    ID of the matched custom metadata type record

cmtSfResultCodeFieldName
    Type: String
    Field that contains the Salesforce result code values. Belongs to the custom metadata type.

CustomMetadataTypeInfo Methods

The following are methods for CustomMetadataTypeInfo.

GatewayErrorResponse Class

Use to respond with an error indication following errors from the PaymentGateway adapter, such as request-forbidden responses, custom validation errors, or expired API tokens.

Namespace

CommercePayments on page 253

Usage

Use GatewayErrorResponse to create an object that stores information about error responses sent by the payment gateway adapter.
Example

If GatewayResponse receives an exception rather than a valid request, it calls GatewayErrorResponse to create an error object with information about the exception.

```apex
global commercepayments.GatewayResponse processRequest(commercepayments.paymentGatewayContext gatewayContext) {
    commercepayments.RequestType requestType = gatewayContext.getPaymentRequestType();
    commercepayments.GatewayResponse response;
    try {
        if (requestType == commercepayments.RequestType.Authorize) {
            response = createAuthResponse((commercepayments.AuthorizationRequest)gatewayContext.getPaymentRequest());
        } else if (requestType == commercepayments.RequestType.Capture) {
            response = createCaptureResponse((commercepayments.CaptureRequest)gatewayContext.getPaymentRequest());
        } else if (requestType == commercepayments.RequestType.ReferencedRefund) {
            response = createRefundResponse((commercepayments.ReferencedRefundRequest)gatewayContext.getPaymentRequest());
        }
        return response;
    } catch(SalesforceValidationException e) {
        commercepayments.GatewayErrorResponse error = new commercepayments.GatewayErrorResponse('400', e.getMessage());
        return error;
    }
}
```

IN THIS SECTION:

GatewayErrorResponse Constructors

GatewayErrorResponse Constructors

The following are constructors for GatewayErrorResponse.

IN THIS SECTION:

GatewayErrorResponse(errorCode, errorMessage)

Constructor to create a GatewayErrorResponse object that accepts errorCode and errorMessage.

GatewayErrorResponse(errorCode, errorMessage)

Constructor to create a GatewayErrorResponse object that accepts errorCode and errorMessage.

Signature

```
global GatewayErrorResponse(String errorCode, String errorMessage)
```
Parameters

errorCode
Type: String
Should match with the HTTP status code to be returned to the user. Here are a few examples.

- If the status code is for a bad request, the errorCode should be 400.
- If the status code is for a forbidden request, errorCode should be 403.
- If errorCode isn’t a valid HTTP status code, a 500 internal server error is returned.

Note: errorCode must have a value, otherwise the platform throws an error.

ererrorMessage
Type: String
The message response to users following an error.

Note: errorMessage must have a value, otherwise the platform throws an error.

GatewayNotificationResponse Class

When the payment gateway sends a notification to the payments platform, the platform responds with a GatewayNotificationResponse indicating whether the platform succeeded or failed at receiving the notification.

Namespace

CommercePayments on page 253

Usage

You must specify the CommercePayments namespace when creating an instance of this class. The constructor of this class takes no arguments. For example:


When an asynchronous payment gateway sends a notification, the gateway requires the platform to acknowledge that it has either succeeded or failed in receiving the notification. Payment gateway adapters use this class to construct the acknowledgment response, which gateways expect for a notification. GatewayNotificationResponse is the return type of the processNotification method.

Example

commercepayments.GatewayNotificationResponse gnr = new commercepayments.GatewayNotificationResponse();
if (saveResult.isSuccess()) {
    system.debug('Notification accepted by platform');
} else {
    system.debug('Errors in the result ' + Blob.valueOf(saveResult.getErrorMessage()));
}
gnr.setStatusCode(200);
IN THIS SECTION:

GatewayNotificationResponse Methods

GatewayNotificationResponse Methods
The following are methods for GatewayNotificationResponse.

IN THIS SECTION:

setResponseBody(responseBody)
Sets the body of the response to the gateway. Some gateways expect the payments platform to acknowledge the notification with a response regardless of whether the notification was accepted.

setStatusCode(statusCode)
Sets the HTTP status code sent to the gateway as part of the payments platform’s response notification.

setResponseBody (responseBody)
Sets the body of the response to the gateway. Some gateways expect the payments platform to acknowledge the notification with a response regardless of whether the notification was accepted.

Signature
global void setResponseBody(Blob responseBody)

Parameters

responseBody
Type: Blob
Common response values include accepted for successfully receiving the response. For example:

```apex
commercepayments.GatewayNotificationResponse gnr = new commercepayments.GatewayNotificationResponse();
if (saveResult.isSuccess()) {
    system.debug('Notification accepted by platform');
} else {
    system.debug('Errors in the result ' + Blob.valueOf(saveResult.getErrorMessage()));
}
gnr.setStatusCode(200);
gnr.setResponseBody(Blob.valueOf('[accepted]'));
return gnr;
```

Return Value
Type: void
setStatusCode(statusCode)
Sets the HTTP status code sent to the gateway as part of the payments platform's response notification.

Signature

    global void setStatusCode(Integer statusCode)

Parameters

    statusCode
        Type: Integer

The status code will vary based on the type of payments platform response. Users should configure their
GatewayNotificationResponse class to account for all values that their payments platform can possibly return. For
example:

```
commercepayments.GatewayNotificationResponse gnr = new
commercepayments.GatewayNotificationResponse();
if (saveResult.isSuccess()) {
    system.debug('Notification accepted by platform');
} else {
    system.debug('Errors in the result '+ Blob.valueOf(saveResult.getErrorMessage()));
}
gnr.setStatusCode(200);
gnr.setResponseBody(Blob.valueOf('[accepted]'));
return gnr;
```

Return Value

Type: void

GatewayResponse Interface

Generic payment gateway response interface. This class extends the CaptureResponse on page 309,
AbstractTransactionResponse on page 265, and AbstractResponse on page 256 classes and inherits all their properties.
It has no unique methods or parameters.

Namespace

CommercePayments on page 253

IN THIS SECTION:

GatewayResponse Example Implementation

GatewayResponse Example Implementation

This is an example implementation of the commercepayments.GatewayResponse interface.

```
/**
 * Abstract function to build gateway response for a Transaction
```
The input is the response from gateway
It creates and returns GatewayResponse from the HttpResponse

```java
public abstract commercepayments.GatewayResponse buildResponse(HttpResponse response);
```

** Function to process transaction requests
Steps involved are:
1. Build HttpRequest with the input Request from gateway context
2. Send request and get the response from gateway
3. Parse the response from gateway and return GatewayResponse

```java
public commercepayments.GatewayResponse execute(){
    HttpRequest req;
    try{
        //Building a new request
        req = buildRequest();
    } catch(PayeezeValidationException e) {
        return getValidationExceptionError(e);
    }
    commercepayments.PaymentsHttp http = new commercepayments.PaymentsHttp();
    HttpResponse res = null;
    try{
        //Sending the request
        res = http.send(req);
    } catch(CalloutException ce) {
        return getCalloutExceptionError(ce);
    }
    try{
        //Parsing the response from gateway
        return buildResponse(res);
    } catch(Exception e) {
        return getParseExceptionError(e);
    }
}
```

For additional context, review the complete Sample Gateway Adapter in the CommercePayments Gateway Reference Implementation.

**NotificationClient Class**

Communicates with the payment platform regarding the gateway’s notification.

**Namespace**

CommercePayments on page 253

**Usage**

Specify the CommercePayments namespace when creating an instance of this class. The constructor of this class takes no arguments. For example:

```java
CommercePayments.NotificationClient ntc = new CommercePayments.NotificationClient();
```
This class is used in asynchronous payment gateway adapters. The notification client contains API for communicating with the payments platform regarding the gateway’s notification. When the gateway sends a notification, the gateway adapter invokes the `record` method in `NotificationClient` to request that the platform updates notification details.

**Example**

The `NotificationSaveResult` class creates a `saveResult` object to store the result of the save request made to the payment gateway.

```java
commercepayments.NotificationSaveResult saveResult = commercepayments.NotificationClient.record(notification);
```

**IN THIS SECTION:**
- NotificationClient Methods

**NotificationClient Methods**

The following are methods for `NotificationClient`.

**record(notification)**

Stores the results of a notification request.

**Signature**

```java
global static commercepayments.NotificationSaveResult record(commercepayments.BaseNotification notification)
```

**Parameters**

- `notification`
  - Type: `BaseNotification` on page 295

**Return Value**

Type: `NotificationSaveResult` on page 328

**NotificationSaveResult Class**

Contains the result of the payment platform’s attempt to record data from the gateway’s notification.

**Namespace**

`CommercePayments` on page 253
Usage

This class is used with asynchronous payments. It is the return type of the NotificationClient.record operation and contains the result of the payment platform’s attempt to save notification details.

The constructor of this class takes no arguments. For example:

```java
CommercePayments.NotificationSaveResult nsr = new CommercePayments.NotificationSaveResult();
```

Example

```java
commercepayments.NotificationSaveResult saveResult = commercepayments.NotificationClient.record(notification);
```

In This Section:

- NotificationSaveResult Methods

NotificationSaveResult Methods

The following are methods for NotificationSaveResult.

In This Section:

- `getErrorMessage()`
  Gets the error message, if any, from the payment platform regarding its attempt to save the notification sent from the payment gateway.

- `getStatusCode()`
  Gets the status code from the payment platform’s attempt to save the notification sent from the payment gateway.

- `isSuccess()`
  Gets the status of whether the payment platform successfully saved the notification sent from the payment gateway.

`getErrorMessage()`

Gets the error message, if any, from the payment platform regarding its attempt to save the notification sent from the payment gateway.

**Signature**

```
global String getErrorMessage()
```

**Return Value**

*Type: String*

`getStatusCode()`

Gets the status code from the payment platform’s attempt to save the notification sent from the payment gateway.

**Signature**

```
global Integer getStatusCode()
```

**Return Value**

*Type: Integer*
Signature

```java
global Integer getStatusCode()
```

Return Value

Type: Integer

`isSuccess()`

Gets the status of whether the payment platform successfully saved the notification sent from the payment gateway.

Signature

```java
global Boolean isSuccess()
```

Return Value

Type: Boolean

NotificationStatus Enum

Shows whether the payments platform successfully received the notification from the gateway.

Usage

When the gateway sends a notification for a payment request, the payments platform delegates the notification request to the gateway adapter. First, the adapter evaluates the signature from the notification request. If the signature is valid, the adapter builds a notification object to store information about the notification. During this process, the adapter sets the `NotificationStatus` to `Failed` or `Success` based on information from the notification request.

Enum Values

The following are the values of the `commercepayments.NotificationStatus` enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Failed</td>
<td>The payments platform couldn't receive the notification due to an error.</td>
</tr>
<tr>
<td>Success</td>
<td>The payments platform received the notification.</td>
</tr>
</tbody>
</table>

PaymentGatewayAdapter Interface

PaymentGatewayAdapters can implement this interface in order to process requests.

Namespace

`CommercePayments` on page 253
IN THIS SECTION:
PaymentGatewayAdapter Methods

PaymentGatewayAdapter Methods
The following are methods for PaymentGatewayAdapter.

IN THIS SECTION:
processRequest(var1)
The entry point for processing payment requests. Returns the response from the payment gateway.

processRequest(var1)
The entry point for processing payment requests. Returns the response from the payment gateway.

Signature
global commercepayments.GatewayResponse processRequest(commercepayments.PaymentGatewayContext var1)

Parameters
var1
Type: commercepayments.PaymentGatewayContext
You can retrieve the request type and the request from the Context object.

Return Value
Type: commercepayments.GatewayResponse
The response from the payment gateway.

PaymentGatewayAsyncAdapter Interface
Implement the interface to allow customers to process payments asynchronously.

Namespace
CommercePayments on page 253

Usage
Implementing an asynchronous adapter also requires the processNotification method from the GatewayNotificationResponse on page 324 class.
Example

global with sharing class SampleAsyncAdapter implements commercepayments.PaymentGatewayAsyncAdapter, commercepayments.PaymentGatewayAdapter {
  global SampleAsyncAdapter() {} 

  global commercepayments.GatewayResponse processRequest(commercepayments.paymentGatewayContext gatewayContext) {
  }

  global commercepayments.GatewayNotificationResponse processNotification(commercepayments.PaymentGatewayNotificationContext gatewayNotificationContext) {
  }
}

IN THIS SECTION:
  PaymentGatewayAsyncAdapter Methods
  PaymentGatewayAsyncAdapter Example Implementation

PaymentGatewayAsyncAdapter Methods

The following are methods for PaymentGatewayAsyncAdapter.

IN THIS SECTION:
  processNotification(paymentGatewayNotificationContext)
  Entry point for processing notifications from payment gateways.

processNotification(paymentGatewayNotificationContext)

Entry point for processing notifications from payment gateways.

Signature

global commercepayments.GatewayNotificationResponse processNotification(commercepayments.PaymentGatewayNotificationContext var1)

Parameters

paymentGatewayNotificationContext
  Type: PaymentGatewayNotificationContext on page 336
  The PaymentGatewayNotificationContext object wraps all the information related to a gateway notification.

Return Value

Type: GatewayNotificationResponse on page 324

When the payment gateway sends a notification to the payments platform, the platform responds with a GatewayNotificationResponse indicating whether the platform succeeded or failed at receiving the notification.
PaymentGatewayAsyncAdapter Example Implementation

This is a sample implementation of the commercepayments.PaymentGatewayAsyncAdapter interface.

global with sharing class AdyenAdapter implements commercepayments.PaymentGatewayAsyncAdapter, commercepayments.PaymentGatewayAdapter {
    global AdyenAdapter() {
    }
    global commercepayments.GatewayResponse processRequest(commercepayments.paymentGatewayContext gatewayContext) {
    }
    global commercepayments.GatewayNotificationResponse processNotification(commercepayments.PaymentGatewayNotificationContext gatewayNotificationContext) {
    }
}

commercepayments.RequestType requestType = gatewayContext.getPaymentRequestType();
if (requestType == commercepayments.RequestType.Capture) {
    req.setEndpoint('/pal/servlet/Payment/v52/capture');
    // Refer to the end of this doc for sample buildCaptureRequest implementation
    body = buildCaptureRequest((commercepayments.CaptureRequest)gatewayContext.getPaymentRequest());
} else if (requestType == commercepayments.RequestType.ReferencedRefund) {
    req.setEndpoint('/pal/servlet/Payment/v52/refund');
    body = buildRefundRequest((commercepayments.ReferencedRefundRequest)gatewayContext.getPaymentRequest());
}

req.setBody(body);
req.setMethod('POST');
commercepayments.PaymentsHttp http = new commercepayments.PaymentsHttp();
HttpResponse res = null;
try {
    res = http.send(req);
} catch(CalloutException ce) {
    commercepayments.GatewayErrorResponse error = new commercepayments.GatewayErrorResponse('500', ce.getMessage());
    return error;
}

if (requestType == commercepayments.RequestType.Capture) {
    // Refer to the end of this doc for sample createCaptureResponse implementation
    response = createCaptureResponse(res);
} else if (requestType == commercepayments.RequestType.ReferencedRefund) {
    response = createRefundResponse(res);
}
return response;

commercepayments.PaymentGatewayNotificationRequest notificationRequest =
gatewayNotificationContext.getPaymentGatewayNotificationRequest();
Blob request = notificationRequest.getRequestBody();
Map<String, Object> jsonReq = (Map<String, Object>)JSON.deserializeUntyped(request.toString());
List<Object> notificationItems = (List<Object>)jsonReq.get('notificationItems');
Map<String, Object> notificationRequestItem =
    ((Map<String, Object>)notificationItems[0]).get('NotificationRequestItem');
Boolean success = Boolean.valueOf(notificationRequestItem.get('success'));
String pspReference = (String)notificationRequestItem.get('pspReference');
String eventCode = (String)notificationRequestItem.get('eventCode');
Double amount = (Double)((Map<String, Object>)notificationRequestItem.get('amount')).get('value');

commercepayments.NotificationStatus notificationStatus = null;
if (success) {
    notificationStatus = commercepayments.NotificationStatus.Success;
} else {
    notificationStatus = commercepayments.NotificationStatus.Failed;
}
commercepayments.BaseNotification notification = null;
if ('CAPTURE'.equals(eventCode)) {
    notification = new commercepayments.CaptureNotification();
} else if ('REFUND'.equals(eventCode)) {
    notification = new commercepayments.ReferencedRefundNotification();
}
notification setStatus(notificationStatus);
notification.setGatewayReferenceNumber(pspReference);
notification.setAmount(amount);

commercepayments.NotificationSaveResult saveResult =
    commercepayments.NotificationClient.record(notification);

commercepayments.GatewayNotificationResponse gnr = new
commercepayments.GatewayNotificationResponse();
if (saveResult.isSuccess()) {
    system.debug('Notification accepted by platform');
} else {
    system.debug('Errors in the result ' + Blob.valueOf(saveResult.getErrorMessage()));
}
gnr.setStatusCode(200);
gnr.setResponseBody(Blob.valueOf('[accepted]'));
return gnr;

PaymentGatewayContext Class
Wraps the information related to a payment request.

Namespace
CommercePayments on page 253

Usage
The constructor of this class takes no arguments. For example:
CommercePayments.PaymentGatewayContext pgc = new CommercePayments.PaymentGatewayContext();

Example

```java
global commercepayments.GatewayResponse processRequest(commercepayments.PaymentGatewayContext gatewayContext) {
    commercepayments.RequestType requestType = gatewayContext.getPaymentRequestType();
    if (requestType == commercepayments.RequestType.Capture) {
        commercepayments.CaptureRequest captureRequest = (commercepayments.CaptureRequest) gatewayContext.getPaymentRequest();
    }
}
```

IN THIS SECTION:
- PaymentGatewayContext Constructors
- PaymentGatewayContext Methods

PaymentGatewayContext Constructors

The following are constructors for PaymentGatewayContext.

IN THIS SECTION:
- PaymentGatewayContext(request, requestType)

Constructor to enable instance creation. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

**PaymentGatewayContext(request, requestType)**

Constructor to enable instance creation. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

Signature

```java
global CommercePayments.PaymentGatewayContext(commercepayments.PaymentGatewayRequest request, String requestType)
```

Parameters

- `request`
  - Type: commercepayments.PaymentGatewayRequest
  - Raw payload. Sensitive attributes are masked to ensure PCI compliance.

- `requestType`
  - Type: String
  - Defines the type of request made to the gateway
PaymentGatewayContext Methods

The following are methods for PaymentGatewayContext.

IN THIS SECTION:

getPaymentRequest()
Returns the payment request object.

gPaymentRequestType()
Returns the payment request type.

getPaymentRequest()

Returns the payment request object.

Signature

global commercepayments.PaymentGatewayRequest getPaymentRequest()

Return Value

Type: PaymentGatewayRequest

getPaymentRequestType()

Returns the payment request type.

Signature

global String getPaymentRequestType()

Return Value

Type: String

PaymentGatewayNotificationContext Class

Wraps the information related to a gateway notification.

Namespace

CommercePayments on page 253

Usage

This class is used with asynchronous payments. It wraps all of the information related to a notification from the payment gateway. The payments platform provides its context to the payment gateway adapters.

The constructor of this class takes no arguments. For example:
CommercePayments.PaymentGatewayNotificationContext pgnc = new CommercePayments.PaymentGatewayNotificationContext();

Example

```java
global commercepayments.GatewayNotificationResponse processNotification(commercepayments.PaymentGatewayNotificationContext
  gatewayNotificationContext) {
  commercepayments.PaymentGatewayNotificationRequest notificationRequest =
  gatewayNotificationContext.getPaymentGatewayNotificationRequest();
}
```

IN THIS SECTION:
  PaymentGatewayNotificationContext Methods

PaymentGatewayNotificationContext Methods

The following are methods for PaymentGatewayNotificationContext.

IN THIS SECTION:
  getPaymentGatewayNotificationRequest()
  Returns the payment gateway’s notification request.

**getPaymentGatewayNotificationRequest()**

Returns the payment gateway’s notification request.

Signature

```java
global commercepayments.PaymentGatewayNotificationRequest
getPaymentGatewayNotificationRequest()
```

Return Value

Type: PaymentGatewayNotificationRequest on page 348

PaymentMethodTokenizationRequest Class

Stores data about a request to tokenize a card payment method. The tokenization process occurs in the payment gateway. This process replaces sensitive customer data, such as a card number or CVV, with unique identification symbols. The symbols are used while the data is handled by Salesforce, the payment gateway, and the customer bank, allowing Salesforce to store the token without storing sensitive customer data.

Namespace

CommercePayments on page 253
Usage

The constructor of this class takes no arguments. For example:

```java
CommercePayments.PaymentMethodTokenizationRequest pmtr = new CommercePayments.PaymentMethodTokenizationRequest();
```

This class holds all the required details about the tokenize request. Gateway adapters read the information in this class while constructing a tokenization JSON request, which is sent to the payment gateway.

Example

The following code is used within your payment gateway adapter Apex class.

Use the `GatewayResponse` class’s `processRequest` method to build responses based on the request type that it receives from an instance of `PaymentGatewayContext` on page 334. If the request type is Tokenize, `GatewayResponse` on page 326 calls the `createTokenizeResponse` method and passes an instance of the `PaymentMethodTokenizationRequest` class. The passed `PaymentMethodTokenizationRequest` object contains the address and cardPaymentMethod information that the payment gateway needs to manage the tokenization process. For example:

```java
public commercepayments.GatewayResponse createTokenizeResponse(commercepayments.PaymentMethodTokenizationRequest tokenizeRequest) {
    commercepayments.PaymentMethodTokenizationResponse tokenizeResponse = new commercepayments.PaymentMethodTokenizationResponse();
    tokenizeResponse.setGatewayTokenEncrypted(encryptedValue);
    tokenizeResponse.setGatewayTokenDetails(tokenDetails);
    tokenizeResponse.setGatewayAvsCode(avsCode);
    return tokenizeResponse;
}
```

Configure the `createTokenizeResponse` method to accept an instance of `PaymentMethodTokenizationRequest`. Then, build an instance of `PaymentMethodTokenizationResponse` based on the values received from the payment gateway.
The `tokenizeResponse` contains the results of the gateway's tokenization process, and if successful, the tokenized value.

IN THIS SECTION:
- PaymentMethodTokenizationRequest Constructors
- PaymentMethodTokenizationRequest Properties
- PaymentMethodTokenizationRequest Methods

**PaymentMethodTokenizationRequest Constructors**
The following are constructors for `PaymentMethodTokenizationRequest`.

IN THIS SECTION:
- PaymentMethodTokenizationRequest(paymentGatewayId)
  Payment gateway ID constructor used with `paymentMethodTokenizationRequest`. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.
- PaymentMethodTokenizationRequest()
  The following are constructors for `PaymentMethodTokenizationRequest`.

### PaymentMethodTokenizationRequest (paymentGatewayId)
Payment gateway ID constructor used with `paymentMethodTokenizationRequest`. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

**Signature**

```java
global PaymentMethodTokenizationRequest(String paymentGatewayId)
```

**Parameters**

- `paymentGatewayId`
  Type: `String`
  The payment method’s payment gateway ID that will be tokenized.

### PaymentMethodTokenizationRequest ()
The following are constructors for `PaymentMethodTokenizationRequest`. 

```java
tokenizeResponse.setGatewayMessage(gatewayMessage);
tokenizeResponse.setGatewayResultCode(resultCode);
tokenizeResponse.setGatewayResultCodeDescription(resultCodeDescription);
tokenizeResponse.setSalesforceResultCodeInfo(resultCodeInfo);
tokenizeResponse.setGatewayDate(system.now());
return tokenizeResponse;
```
Signature

```csharp
public PaymentMethodTokenizationRequest()
```

PaymentMethodTokenizationRequest Properties

The following are properties for `PaymentMethodTokenizationRequest`.

IN THIS SECTION:

- `address`
  - The card payment method address to be tokenized.
- `cardPaymentMethod`
  - The card payment method containing data to be tokenized.

`address`

The card payment method address to be tokenized.

Signature

```csharp
public commercepayments.AddressRequest address {get; set;}
```

Property Value

Type: `AddressRequest` on page 259

`cardPaymentMethod`

The card payment method containing data to be tokenized.

Signature

```csharp
public commercepayments.CardPaymentMethodRequest cardPaymentMethod {get; set;}
```

Property Value

Type: `CardPaymentMethodRequest` on page 315

PaymentMethodTokenizationRequest Methods

The following are methods for `PaymentMethodTokenizationRequest`.

IN THIS SECTION:

- `equals(obj)`
  - Maintains the integrity of lists of type `PaymentMethodTokenizationRequest` by determining the equality of external objects in a list. This method is dynamic and is based on the equals method in Java.
hashCode()  
Maintains the integrity of lists of type PaymentMethodTokenizationRequest by determining the uniqueness of the external object records in a list.

toString()  
Converts a date to a string.

equals(obj)  
Maintains the integrity of lists of type PaymentMethodTokenizationRequest by determining the equality of external objects in a list. This method is dynamic and is based on the equals method in Java.

Signature

```java
global Boolean equals(Object obj)
```

Parameters

- `obj`  
  Type: Object  
  External object whose key is to be validated.

Return Value

Type: Boolean

hashCode()  
Maintains the integrity of lists of type PaymentMethodTokenizationRequest by determining the uniqueness of the external object records in a list.

Signature

```java
global Integer hashCode()
```

Return Value

Type: Integer

toString()  
Converts a date to a string.

Signature

```java
global String toString()
```

Return Value

Type: String
PaymentMethodTokenizationResponse Class

Gateway response sent by payment gateway adapters for the payment method tokenization request. The response includes the payment method’s token ID value.

Namespace

CommercePayments on page 253

Usage

The constructor of this class takes no arguments. For example:

```java
CommercePayments.PaymentMethodTokenizationResponse pmtr = new CommercePayments.PaymentMethodTokenizationResponse();
```

After the payment gateway processes a tokenization request, the fields of PaymentMethodTokenizationResponse receive and store information from the gateway’s response. The gateway’s response shows whether the tokenization request was successful, the token value, and any additional messages or information about the tokenization process. You can then pass an instance of PaymentMethodTokenizationResponse to an authorization response or a sale response. This class is mapped to a response class in the Java layer.

Example

This constructor builds a new instance of the PaymentMethodTokenizationResponse class.

```java
commercepayments.PaymentMethodTokenizationResponse tokenizeResponse = new commercepayments.PaymentMethodTokenizationResponse();
```

PaymentMethodTokenizationResponse contains only setter methods. Each setter accepts a value from the payment gateway and use it to set an attribute of PaymentMethodTokenizationResponse.

The most important method in PaymentMethodTokenizationResponse is setGatewayTokenEncrypted, which uses Salesforce encryption to set an encrypted token value for a payment method. The setGatewayTokenEncrypted method is available in Salesforce API v52.0 and later. We recommend using it to ensure your tokenized payment method values are encrypted and secure. While the setGatewayToken method (available in earlier API versions) also returns a payment method token, the tokenized value isn’t encrypted.

If the instantiated class already has a gateway token, setGatewayTokenEncrypted throws an error.

```java
/** @description Method to set Gateway token to persist in Encrypted Text */
global void setGatewayTokenEncrypted(String gatewayTokenEncrypted) {
    if (gatewayTokenSet) {
        throwTokenError();
    }
    this.delegate.setGatewayTokenEncrypted(gatewayTokenEncrypted);
    gatewayTokenEncryptedSet = true;
}
```

A typical instantiation of PaymentMethodTokenizationResponse sets the encrypted gateway token alongside the other tokenization response values sent by the gateway.

```java
public commercepayments.GatewayResponse createTokenizeResponse(commercepayments.PaymentMethodTokenizationRequest tokenizeRequest)
```
After you've built a `PaymentMethodTokenizationResponse` object and set the encrypted gateway token, pass the object to the `setPaymentMethodTokenizationResponse` method of an authorization response or a sale response.

**Authorization Response**

```java
public commercepayments.GatewayResponse createAuthResponse(commercepayments.AuthorizationRequest authRequest) {
    commercepayments.PaymentMethodTokenizationResponse paymentMethodTokenizationResponse = new commercepayments.PaymentMethodTokenizationResponse();
    if(authRequest.amount!=null )
    {
        authResponse.setAmount(authRequest.amount);
    }
    else
    {
        throw new SalesforceValidationException('Required Field Missing : Amount');
    }
    authResponse.setGatewayResultCode('00');
    authResponse.setGatewayResultCodeDescription('Transaction Normal');
    authResponse.setGatewayAuthCode('SF'+getRandomNumber(6));
    authResponse.setGatewayReferenceNumber(getRandomNumber(10));
    authResponse.setSalesforceResultCodeInfo(SUCCESS_SALESFORCE_RESULT_CODE_INFO);
    paymentMethodTokenizationResponse.setGatewayTokenEncrypted(gatewayTokenEncrypted);
    authResponse.setPaymentMethodTokenizationResponse(paymentMethodTokenizationResponse);
    authResponse.setGatewayDate(system.now());
    return authResponse;
}
```

**Sale Response**

```java
public commercepayments.GatewayResponse createSaleResponse(commercepayments.SaleRequest saleRequest) {
    commercepayments.SaleResponse saleResponse = new commercepayments.SaleResponse();
    return saleResponse;
}
```
commercepayments.PaymentMethodTokenizationResponse

paymentMethodTokenizationResponse = new commercepayments.PaymentMethodTokenizationResponse();

if(saleRequest.amount!=null )
{
    saleResponse.setAmount(saleRequest.amount);
}
else
{
    throw new SalesforceValidationException('Required Field Missing : Amount');
}

system.debug('Response - success');
saleResponse.setGatewayDate(system.now());
saleResponse.setGatewayResultCode('00');
saleResponse.setGatewayResultCodeDescription('Transaction Normal');
saleResponse.setGatewayReferenceNumber('SF'+getRandomNumber(6));
saleResponse.setSalesforceResultCodeInfo(SUCCESS_SALESFORCE_RESULT_CODE_INFO);

paymentMethodTokenizationResponse.setGatewayTokenEncrypted(gatewayTokenEncrypted);
saleResponse.setPaymentMethodTokenizationResponse(paymentMethodTokenizationResponse);

return saleResponse;

IN THIS SECTION:
PaymentMethodTokenizationResponse Methods

PaymentMethodTokenizationResponse Methods
The following are methods for PaymentMethodTokenizationResponse.

IN THIS SECTION:

setGatewayAvsCode(gatewayAvsCode)
Sets the AVS (address verification system) result code information that the gateway returned. Maximum length of 64 characters.

setGatewayDate(gatewayDate)
Sets the date that the tokenization occurred. Some gateways don't send this value.

setGatewayMessage(gatewayMessage)
Sets error messages that the gateway returned for the tokenization request. Maximum length of 255 characters.

setGatewayResultCode(gatewayResultCode)
Sets a gateway-specific result code. The code may be mapped to a Salesforce-specific result code. Maximum length of 64 characters.

setGatewayResultCodeDescription(gatewayResultCodeDescription)
Sets a description of the gateway-specific result code that a payment gateway returned. Maximum length of 1000 characters.
setGatewayToken(gatewayToken)
Sets the gateway token value that the gateway returned.

setGatewayTokenDetails(gatewayTokenDetails)
Sets any additional information that the gateway returned about the payment token.

setGatewayTokenEncrypted(gatewayTokenEncrypted)
Sets the value of the gatewayTokenEncrypted field on a CardPaymentMethod or DigitalWallet object.

setSalesforceResultCodeInfo(salesforceResultCodeInfo)
Sets the Salesforce-specific result code information. Payment gateways have many response codes for payment calls. Salesforce uses the result code information to map payment gateway codes to a predefined set of standard Salesforce result codes.

setGatewayAvsCode (gatewayAvsCode)
Sets the AVS (address verification system) result code information that the gateway returned. Maximum length of 64 characters.

Signature

```java
global void setGatewayAvsCode(String gatewayAvsCode)
```

Parameters

gatewayAvsCode
Type: String
Used to verify the address mapped to a payment method when the payments platform requests tokenization from the payment gateway.

Return Value
Type: void

setGatewayDate (gatewayDate)
Sets the date that the tokenization occurred. Some gateways don’t send this value.

Signature

```java
global void setGatewayDate(Datetime gatewayDate)
```

Parameters

gatewayDate
Type: Datetime

Return Value
Type: void

setGatewayMessage (gatewayMessage)
Sets error messages that the gateway returned for the tokenization request. Maximum length of 255 characters.
**Signature**

global void setGatewayMessage(String gatewayMessage)

**Parameters**

gatewayMessage
  Type: String

**Return Value**

Type: void

---

**setGatewayResultCode (gatewayResultCode)**

Sets a gateway-specific result code. The code may be mapped to a Salesforce-specific result code. Maximum length of 64 characters.

**Signature**

global void setGatewayResultCode(String gatewayResultCode)

**Parameters**

gatewayResultCode
  Type: String
  Gateway-specific result code. Must be used to map a Salesforce-specific result code.

**Return Value**

Type: void

---

**setGatewayResultCodeDescription (gatewayResultCodeDescription)**

Sets a description of the gateway-specific result code that a payment gateway returned. Maximum length of 1000 characters.

**Signature**

global void setGatewayResultCodeDescription(String gatewayResultCodeDescription)

**Parameters**

gatewayResultCodeDescription
  Type: String
  Provides additional information about the result code and why the gateway returned the specific code. Descriptions will vary between different gateways.

**Return Value**

Type: void
**setGatewayToken (gatewayToken)**
Sets the gateway token value that the gateway returned.

**Signature**

```java
global void setGatewayToken(String gatewayToken)
```

**Parameters**

- **gatewayToken**
  - Type: `String`
  - The gateway token that the payment gateway sends following a tokenization request.
  - For the CardPaymentMethod and DigitalWallet objects, use the `gatewayTokenEncrypted` parameter, which encrypts the token value.

**Return Value**

Type: `void`

**setGatewayTokenDetails (gatewayTokenDetails)**
Sets any additional information that the gateway returned about the payment token.

**Signature**

```java
global void setGatewayTokenDetails(String gatewayTokenDetails)
```

**Parameters**

- **gatewayTokenDetails**
  - Type: `String`

**Return Value**

Type: `void`

**setGatewayTokenEncrypted (gatewayTokenEncrypted)**
Sets the value of the `gatewayTokenEncrypted` field on a CardPaymentMethod or DigitalWallet object.

**Signature**

```java
global void setGatewayTokenEncrypted(String gatewayTokenEncrypted)
```

**Parameters**

- **gatewayTokenEncrypted**
  - Type: `String`
The gateway token that the payment gateway sends following a tokenization request. Salesforce Payments uses Salesforce encryption to encrypt the token value.

**Return Value**

Type: void

**setSalesforceResultCodeInfo (salesforceResultCodeInfo)**

Sets the Salesforce-specific result code information. Payment gateways have many response codes for payment calls. Salesforce uses the result code information to map payment gateway codes to a predefined set of standard Salesforce result codes.

**Signature**


**Parameters**

`salesforceResultCodeInfo`  
Type: `SalesforceResultCodeInfo` on page 377  
Description of the Salesforce result code value.

**Return Value**

Type: void

**PaymentGatewayNotificationRequest Class**

Contains the notification request data from the gateway.

**Namespace**

`CommercePayments` on page 253

**Usage**

When the payment gateway sends a notification for a payment request, the payments platform sends the notification request to the gateway adapter. If the notification payload contains an `eventCode` of CAPTURE, the adapter constructs a `CaptureNotification`. If the notification payload contains an `eventCode` of REFUND, the adapter constructs a `ReferencedRefundNotification`. If the notification payload contains `eventCode` of AUTHORIZATION, the adapter constructs a `GatewayNotificationResponse`.

You can obtain a notification request from `PaymentGatewayNotificationContext` on page 336 by invoking its `getPaymentGatewayNotificationRequest` method.
Example

global commercepayments.GatewayNotificationResponse
  processNotification(commercepayments.PaymentGatewayNotificationContext
  gatewayNotificationContext) {
    commercepayments.PaymentGatewayNotificationRequest notificationRequest =
    gatewayNotificationContext.getPaymentGatewayNotificationRequest();
  }

IN THIS SECTION:
  PaymentGatewayNotificationRequest Properties
  PaymentGatewayNotificationRequest Methods

PaymentGatewayNotificationRequest Properties
The following are properties for PaymentGatewayNotificationRequest.

IN THIS SECTION:
  requestBody
  Signature

requestBody

Body of the notification request sent by the payment gateway.

Signature

global Blob requestBody {get; set;}

Property Value
Type: Blob

PaymentGatewayNotificationRequest Methods
The following are methods for PaymentGatewayNotificationRequest.

IN THIS SECTION:
  getHeaders()
  getRequestBody()

getHeaders()

Gets HTTP headers from the notification request sent by the payment gateway.

getRequestBody()

Stores the notification request body information from the payment gateway’s notification request.

getHeaders ()

Gets HTTP headers from the notification request sent by the payment gateway.
Signature

```java
global Map<String, String> getHeaders()
```

Return Value

Type: Map<String, String>

**getRequestBody()**

Stores the notification request body information from the payment gateway's notification request.

Signature

```java
global Blob getRequestBody()
```

Return Value

Type: Blob

**PaymentsHttp Class**

Makes an HTTP request to start the interaction with the payment gateway.

**Namespace**

*CommercePayments* on page 253

**Usage**

You must specify the *CommercePayments* namespace when creating an instance of this class. The constructor of this class takes no arguments. For example:

```java
CommercePayments.PaymentsHttp payhttp = new CommercePayments.PaymentsHttp();
```

**IN THIS SECTION:**

- PaymentsHttp Methods
- PaymentsHttp Constructors

**PaymentsHttp Methods**

The following are methods for *PaymentsHttp*. All methods are instance methods.

**IN THIS SECTION:**

- send(Request)
  Sends an HttpRequest and returns the response.
**send (Request)**
Sends an HttpRequest and returns the response.

**Signature**

```
global HttpResponse send(HttpRequest request)
```

**Parameters**

`request`
Type: `System.HttpRequest`

**Return Value**
Type: `System.HttpResponse`

**PaymentsHttp Constructors**
The following are constructors for `PaymentsHttp`.

**IN THIS SECTION:**

```
PaymentsHttp()
```
Initiates an HTTP request and response.

**PaymentsHttp ()**
Initiates an HTTP request and response.

**Signature**

```
global PaymentsHttp()
```

**RefundRequest Class**
Sends data related to a refund to the payment gateway adapter.

**Namespace**

`CommercePayments` on page 351

**Usage**
The constructor of this class takes no arguments. For example:

```
CommercePayments.RefundRequest rrq = new CommercePayments.RefundRequest();
```
Example

```java
commercepayments.ReferencedRefundRequest refundRequest = new
commercepayments.ReferencedRefundRequest(80, pmt.id);
```

IN THIS SECTION:
- RefundRequest Methods

**RefundRequest Methods**

The following are methods for RefundRequest.

IN THIS SECTION:
- `equals(obj)`
- `hashCode()`

**equals(obj)**

Maintains the integrity of lists of type `RefundRequest` by determining the equality of external objects in a list. This method is dynamic and is based on the equals method in Java.

**hashCode()**

Maintains the integrity of lists of type `RefundRequest` by determining the uniqueness of the external object records in a list.

**equals(obj)**

Maintains the integrity of lists of type `RefundRequest` by determining the equality of external objects in a list. This method is dynamic and is based on the equals method in Java.

**Signature**

```java
global Boolean equals(Object obj)
```

**Parameters**

- `obj`
  - Type: Object

**Return Value**

- Type: Boolean

**hashCode()**

Maintains the integrity of lists of type `RefundRequest` by determining the uniqueness of the external object records in a list.

**Signature**

```java
global Integer hashCode()
```

**Return Value**

- Type: Integer

352
ReferencedRefundNotification Class

When a payment gateway sends a notification for a refund transaction, the payment gateway adapter creates the ReferencedRefundNotification object to store information about notification.

Namespace

CommercePayments on page 253

Usage

This class is used with asynchronous payments. When a payment gateway sends a notification for a refund transaction, the gateway adapter creates an object of type ReferencedRefundNotification to populate the respective values.

The constructor of this class takes no arguments. For example:

```java
CommercePayments.ReferencedRefundNotification rrn = new CommercePayments.ReferencedRefundNotification();
```

Example

```java
commercepayments.NotificationStatus notificationStatus = null;
if (success) {
    notificationStatus = commercepayments.NotificationStatus.Success;
} else {
    notificationStatus = commercepayments.NotificationStatus.Failed;
}
commercepayments.BaseNotification notification = null;
if ('CAPTURE'.equals(eventCode)) {
    notification = new commercepayments.CaptureNotification();
} else if ('REFUND'.equals(eventCode)) {
    notification = new commercepayments.ReferencedRefundNotification();
}
```

ReferencedRefundNotification Methods

The following are methods for ReferencedRefundNotification.

setAmount(amount)
Sets the transaction amount. Can be positive, negative, or zero.

setGatewayDate(gatewayDate)
Sets the date that communication for the refund notification occurred with the payment gateway.

setGatewayMessage(gatewayMessage)
Sets information or messages that the gateway returned.
**setGatewayReferenceDetails(gatewayReferenceDetails)**
Sets the payment gateway’s reference details.

**setGatewayReferenceNumber(gatewayReferenceNumber)**
Sets the payment gateway’s reference number.

**setGatewayResultCode(gatewayResultCode)**
Sets the payment gateway’s result code.

**setGatewayResultCodeDescription(gatewayResultCodeDescription)**
Sets the payment gateway’s result code description.

**setId(id)**
Sets the ID of a notification sent by the payment gateway.

**setSalesforceResultCodeInfo(salesforceResultCodeInfo)**
Sets Salesforce result code information.

**setStatus(status)**
Sets the notification status value on the notification object.

---

**setAmount(amount)**
Sets the transaction amount. Can be positive, negative, or zero.

**Signature**

```java
global void setAmount(Double amount)
```

**Parameters**

`amount`
Type: `Double`
The amount to be debited or captured.

**Return Value**

Type: void

---

**setGatewayDate(gatewayDate)**
Sets the date that communication for the refund notification occurred with the payment gateway.

**Signature**

```java
global void setGatewayDate(Datetime gatewayDate)
```

**Parameters**

`gatewayDate`
Type: `Datetime`
The date that communication happened with the gateway.
Return Value
Type: void

*setGatewayMessage*(gatewayMessage)
Sets information or messages that the gateway returned.

**Signature**
```
global void setGatewayMessage(String gatewayMessage)
```

**Parameters**

*gatewayMessage*  
Type: *String*

Return Value
Type: void

*setGatewayReferenceDetails*(gatewayReferenceDetails)
Sets the payment gateway's reference details.

**Signature**
```
global void setGatewayReferenceDetails(String gatewayReferenceDetails)
```

**Parameters**

*gatewayReferenceDetails*  
Type: *String*

  Provides information about the gateway communication.

Return Value
Type: void

*setGatewayReferenceNumber*(gatewayReferenceNumber)
Sets the payment gateway's reference number.

**Signature**
```
global void setGatewayReferenceNumber(String gatewayReferenceNumber)
```

**Parameters**

*gatewayReferenceNumber*  
Type: *String*
Unique transaction ID created by the payment gateway.

*Return Value*
Type: void

**setGatewayResultCode** *gatewayResultCode*
Sets the payment gateway's result code.

*Signature*

```java
global void setGatewayResultCode(String gatewayResultCode)
```

*Parameters*

`gatewayResultCode`
Type: `String`
The gateway result code. You must map this to a Salesforce-specific result code.

*Return Value*
Type: void

**setGatewayResultCodeDescription** *gatewayResultCodeDescription*
Sets the payment gateway's result code description.

*Signature*

```java
global void setGatewayResultCodeDescription(String gatewayResultCodeDescription)
```

*Parameters*

`gatewayResultCodeDescription`
Type: `String`
Description of the gateway result code. Provides additional context about the result code.

*Return Value*
Type: void

**setId** *id*
Sets the ID of a notification sent by the payment gateway.

*Signature*

```java
global void setId(String id)
```
Parameters

$id$
Type: String

Return Value
Type: void

`setSalesforceResultCodeInfo(salesforceResultCodeInfo)`
Sets Salesforce result code information.

Signature


Parameters

`salesforceResultCodeInfo`
Type: `SalesforceResultCodeInfo` on page 377
Description of the Salesforce result code value.

Return Value
Type: void

`setStatus(status)`
Sets the notification status value on the notification object.

Signature

`global void setStatus(commercepayments.NotificationStatus status)`

Parameters

`status`
Type: `NotificationStatus` on page 330
Indicates whether the payments platform successfully received the notification from the payment gateway.

Return Value
Type: void

`ReferencedRefundRequest`
Access information about the referenced refund requests. Extends the `RefundRequest` class.
Example

```java
global commercepayments.GatewayResponse processRequest(commercepayments.PaymentGatewayContext gatewayContext) {
  commercepayments.RequestType requestType = gatewayContext.getPaymentRequestType();
  if (requestType == commercepayments.RequestType.ReferencedRefund) {
    commercepayments.*ReferencedRefundRequest* refundRequest =
    (commercepayments.*ReferencedRefundRequest*) gatewayContext.getPaymentRequest();
  }
}
```

IN THIS SECTION:
- ReferencedRefundRequest Constructors
- ReferencedRefundRequest Properties
- ReferencedRefundRequest Methods

ReferencedRefundRequest Constructors

The following are constructors for ReferencedRefundRequest.

IN THIS SECTION:
- ReferencedRefundRequest(amount, paymentId)
  
  This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

ReferencedRefundRequest(amount, paymentId)

This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

Parameters

- amount
  
  Type: Double
  
  The amount to be debited or captured.

- paymentId
  
  Type: String
  
  The payment record.

ReferencedRefundRequest Properties

The following are properties for ReferencedRefundRequest.
IN THIS SECTION:

**PaymentId**
References a payment object.

**accountId**
References an account.

**amount**
References an amount.

---

**PaymentId**
References a payment object.

**Property Value**
Type: String

**accountId**
References an account.

**Property Value**
Type: String

**amount**
References an amount.

**Property Value**
Type: Double

---

**ReferencedRefundRequest Methods**
The following are methods for ReferencedRefundRequest.

---

**ReferencedRefundResponse Class**
The payment gateway adapter sends this response for the ReferencedRefund request type.

**Namespace**
CommercePayments on page 253

**Usage**
The constructor of this class takes no arguments. For example:
CommercePayments.ReferencedRefundResponse refr = new CommercePayments.ReferencedRefundResponse();

IN THIS SECTION:
ReferencedRefundResponse Methods

ReferencedRefundResponse Methods
The following are methods for ReferencedRefundResponse.

IN THIS SECTION:
  setAmount(amount)
  Sets the transaction amount. The value must be a positive number.
  setGatewayAvsCode(gatewayAvsCode)
  Sets the payment gateway’s address verification system (AVS) code.
  setGatewayDate(gatewayDate)
  Sets the payment gateway’s date.
  setGatewayMessage(gatewayMessage)
  Sets information or messages that the gateway returned.
  setGatewayReferenceDetails(gatewayReferenceDetails)
  Sets the payment gateway’s reference details.
  setGatewayReferenceNumber(gatewayReferenceNumber)
  Sets the payment gateway’s reference number.
  setGatewayResultCode(gatewayResultCode)
  Sets the payment gateway’s result code.
  setGatewayResultCodeDescription(gatewayResultCodeDescription)
  Sets the payment gateway’s result code description.
  setSalesforceResultCodeInfo(salesforceResultCodeInfo)
  Set the Salesforce result code info.

setAmount(amount)
Sets the transaction amount. The value must be a positive number.

Signature

global void setAmount(Double amount)

Parameters

amount
  Type: Double
  The amount to be debited or captured.

360
Return Value
Type: void

**setGatewayAvsCode(gatewayAvsCode)**
Sets the payment gateway's address verification system (AVS) code.

**Signature**

```java
global void setGatewayAvsCode(String gatewayAvsCode)
```

**Parameters**

- **gatewayAvsCode**
  - Type: String
  - Code sent by gateways that use an address verification system.

Return Value
Type: void

**setGatewayDate(gatewayDate)**
Sets the payment gateway's date.

**Signature**

```java
global void setGatewayDate(Datetime gatewayDate)
```

**Parameters**

- **gatewayDate**
  - Type: Datetime
  - Date and time of the gateway communication.

Return Value
Type: void

**setGatewayMessage(gatewayMessage)**
Sets information or messages that the gateway returned.

**Signature**

```java
global void setGatewayMessage(String gatewayMessage)
```
Parameters

`gatewayMessage`
Type: String
Information or error messages returned by the gateway.

Return Value
Type: void

`setGatewayReferenceDetails (gatewayReferenceDetails)`
Sets the payment gateway's reference details.

Signature

`global void setGatewayReferenceDetails(String gatewayReferenceDetails)`

Parameters

`gatewayReferenceDetails`
Type: String
Information about the gateway communication.

Return Value
Type: void

`setGatewayReferenceNumber (gatewayReferenceNumber)`
Sets the payment gateway's reference number.

Signature

`global void setGatewayReferenceNumber(String gatewayReferenceNumber)`

Parameters

`gatewayReferenceNumber`
Type: String
Unique transaction ID created by the payment gateway.

Return Value
Type: void

`setGatewayResultCode (gatewayResultCode)`
Sets the payment gateway's result code.
global void setGatewayResultCode(String gatewayResultCode)

Parameters

gatewayResultCode
    Type: String
    The gateway result code. Must be mapped to a Salesforce result code.

Return Value
Type: void

setGatewayResultCodeDescription(String gatewayResultCodeDescription)
Sets the payment gateway's result code description.

Signature

global void setGatewayResultCodeDescription(String gatewayResultCodeDescription)

Parameters

gatewayResultCodeDescription
    Type: String
    Description of the GatewayResultCode. Provides more information about the result code returned by the gateway.

Return Value
Type: void

setSalesforceResultCodeInfo(SalesforceResultCodeInfo)
Set the Salesforce result code info.

Signature

global void setSalesforceResultCodeInfo(commercepayments.SalesforceResultCodeInfo)

Parameters

salesforceResultCodeInfo
    Type: commercepayments.SalesforceResultCodeInfo on page 377
    Describes the Salesforce result code value.

Return Value
Type: void
RequestType Enum

Defines the type of payment transaction request made to the payment gateway.

Enum Values

The following are the values of the commercepayments.RequestType enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Authorize</td>
<td>Payment authorization request</td>
</tr>
<tr>
<td>Capture</td>
<td>Payment capture request</td>
</tr>
<tr>
<td>ReferencedRefund</td>
<td>Payment refund request</td>
</tr>
<tr>
<td>Sale</td>
<td>Sale request</td>
</tr>
<tr>
<td>Tokenization</td>
<td>Payment tokenization request</td>
</tr>
</tbody>
</table>

SaleApiPaymentMethodRequest Class

Sends data related to a card payment method to a gateway adapter during a sale service call.

Namespace

CommercePayments on page 253

Usage

This class holds information about a payment method that was used for a Sale request. SaleApiPaymentMethodRequest contains all the possible payment methods as fields, but only one value is populated for a given request. Gateway adapters use this class when constructing a sale request. The object of this class is obtained through the paymentMethod field on the SaleRequest object.

Example: This code sample retrieves the SaleApiPaymentMethodRequest object from the SaleRequest class.

```
commercepayments.SaleApiPaymentMethodRequest paymentMethod = saleRequest.paymentMethod;
```

IN THIS SECTION:

SaleApiPaymentMethodRequest Constructors
SaleApiPaymentMethodRequest Properties
SaleApiPaymentMethodRequest Methods
SaleApiPaymentMethodRequest Constructors

The following are constructors for SaleApiPaymentMethodRequest.

IN THIS SECTION:

SaleApiPaymentMethodRequest(cardPaymentMethodRequest)
Sends data related to a card payment method to a gateway adapter during a sale service call.

SaleApiPaymentMethodRequest()
Constructor for building a sale payment method request. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

SaleApiPaymentMethodRequest (cardPaymentMethodRequest)
Sends data related to a card payment method to a gateway adapter during a sale service call.

Signature

global SaleApiPaymentMethodRequest (commercepayments.CardPaymentMethodRequest cardPaymentMethodRequest)

Parameters

cardPaymentMethodRequest
Type: CardPaymentMethodRequest on page 315

SaleApiPaymentMethodRequest ()
Constructor for building a sale payment method request. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

Signature

global SaleApiPaymentMethodRequest ()

SaleApiPaymentMethodRequest Properties

The following are properties for SaleApiPaymentMethodRequest.

IN THIS SECTION:

cardPaymentMethod
Contains details of the card used in a payment method.

cardPaymentMethod
Contains details of the card used in a payment method.
Signature

global commercepayments.CardPaymentMethodRequest cardPaymentMethod {get; set;}

Property Value

Type: CardPaymentMethodRequest on page 315

SaleApiPaymentMethodRequest Methods

The following are methods for SaleApiPaymentMethodRequest.

IN THIS SECTION:

equals(obj)
Maintains the integrity of lists of type SaleApiPaymentMethodRequest by determining the equality of external objects in a list. This method is dynamic and is based on the equals method in Java.

hashCode()
Maintains the integrity of lists of type SaleApiPaymentMethodRequest by determining the uniqueness of the external object records in a list.

toString()
Converts a date to a string.

equals(obj)
Maintains the integrity of lists of type SaleApiPaymentMethodRequest by determining the equality of external objects in a list. This method is dynamic and is based on the equals method in Java.

Signature

global Boolean equals(Object obj)

Parameters

obj
Type: Object

Return Value

Type: Boolean

hashCode()
Maintains the integrity of lists of type SaleApiPaymentMethodRequest by determining the uniqueness of the external object records in a list.

Signature

global Integer hashCode()
Return Value
Type: Integer

toString()
Converts a date to a string.

Signature
global String toString()

Return Value
Type: String

SaleRequest Class
Stores information about a sales request.

Namespace
CommercePayments on page 253

Usage
This class holds all the required details about a sale request. Gateway adapters read the fields of this class object while constructing a sale JSON request that is sent to the payment gateway. The object of this class is made available through commercepayments.paymentGatewayContext by calling getPaymentRequest().

Example
This code sample retrieves the SaleRequest object from the PaymentGatewayContext class.

```java
commercepayments.SaleRequest = (commercepayments.SaleRequest)gatewayContext.getPaymentRequest()
```

IN THIS SECTION:
  SaleRequest Constructors
  SaleRequest Properties
  SaleRequest Methods

SaleRequest Constructors
The following are constructors for SaleRequest.
IN THIS SECTION:

**SaleRequest(amount)**

Constructor for defining an amount for the sale request. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

**SaleRequest(amount)**

Constructor for defining an amount for the sale request. This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

**Signature**

```
global SaleRequest(Double amount)
```

**Parameters**

- **amount**
  - Type: `Double`
  - Amount of the sale request.

**SaleRequest Properties**

The following are properties for `SaleRequest`.

IN THIS SECTION:

- **accountId**
  - Customer account ID for the sale request.
- **amount**
  - Amount of the sale request. Can be positive only.
- **comments**
  - Additional information about the sale request.
- **currencyIsoCode**
  - Currency code for the sale request.
- **paymentMethod**
  - Payment method used in the sale request.

**accountId**

Customer account ID for the sale request.

**Signature**

```
global String accountId {get; set;}
```
Property Value
Type: String

**amount**
Amount of the sale request. Can be positive only.

Signature
```csharp
global Double amount {get; set;}
```

Property Value
Type: Double

**comments**
Additional information about the sale request.

Signature
```csharp
global String comments {get; set;}
```

Property Value
Type: String

**currencyIsoCode**
Currency code for the sale request.

Signature
```csharp
global String currencyIsoCode {get; set;}
```

Property Value
Type: String

**paymentMethod**
Payment method used in the sale request.

Signature
```csharp
global commercepayments.SaleApiPaymentMethodRequest paymentMethod {get; set;}
```

Property Value
Type: SaleApiPaymentMethodRequest on page 364
**SaleRequest Methods**

The following are methods for SaleRequest.

**IN THIS SECTION:**
- **equals(obj)**
  Compares this object with the specified object and returns `true` if both objects are equal; otherwise, returns `false`.
- **hashCode()**
  Maintains the integrity of lists of type SaleRequest by determining the uniqueness of the external object records in a list.
- **toString()**
  Converts a date to a string.

**equals(obj)**

Compares this object with the specified object and returns `true` if both objects are equal; otherwise, returns `false`.

**Signature**

```java
global Boolean equals(Object obj)
```

**Parameters**

- **obj**
  Type: Object

**Return Value**

Type: Boolean

**hashCode()**

Maintains the integrity of lists of type SaleRequest by determining the uniqueness of the external object records in a list.

**Signature**

```java
global Integer hashCode()
```

**Return Value**

Type: Integer

**toString()**

Converts a date to a string.

**Signature**

```java
global String toString()
```
Return Value
Type: String

SaleResponse Class
Response sent by payment gateway adapters for a sales service.

Namespace
CommercePayments on page 253

Usage
The constructor of this class takes no arguments. For example:

CommercePayments.SaleResponse slr CommercePayments.SaleResponse();

This class contains details about a customer card that was used as a payment method for Authorization, Sale, or Tokenization request. The gateway adapter reads the fields of this class while constructing a transaction JSON request, which it then sends to the payment gateway. The object of this class is made available by the cardPaymentMethod field in SaleApiPaymentMethodRequest on page 364 and AuthApiPaymentMethodRequest on page 270.

Example
This code sample builds a SaleResponse object.

```java
commercepayments.SaleResponse saleResponse = new commercepayments.SaleResponse();
saleResponse.setGatewayReferenceDetails("refDetailString");
saleResponse.setGatewayResultCode("res_code");
saleResponse.setGatewayResultCodeDescription(""");
saleResponse.setGatewayReferenceNumber("");
saleResponse.setSalesforceResultCodeInfo(getSalesforceResultCodeInfo(commercepayments.SalesforceResultCode.SUCCESS.name()));
```

IN THIS SECTION:
SaleResponse Methods

SaleResponse Methods
The following are methods for SaleResponse.

IN THIS SECTION:
setAmount(amount)
Sets the transaction amount. Must be a non-negative value.

setGatewayAvsCode(gatewayAvsCode)
Sets the AVS (address verification system) result code information that the gateway returned. Maximum length of 64 characters.

setGatewayDate(gatewayDate)
Sets the date that the sale occurred. Some gateways don't send this value.
setGatewayMessage(gatewayMessage)
Sets error messages that the gateway returned for the sale request. Maximum length of 255 characters.

setGatewayReferenceDetails(gatewayReferenceDetails)
Sets additional data that you can use for subsequent sales. You can use any data that isn’t normalized in financial entities. This field has a maximum length of 1000 characters and can store data as JSON or XML.

setGatewayReferenceNumber(gatewayReferenceNumber)
Sets the unique gateway reference number for the transaction that the gateway returned. Maximum length of 255 characters.

setGatewayResultCode(gatewayResultCode)
Sets a gateway-specific result code. The code may be mapped to a Salesforce-specific result code. Maximum length of 64 characters.

setGatewayResultCodeDescription(gatewayResultCodeDescription)
Sets a description of the gateway-specific result code that a payment gateway returned. Maximum length of 1000 characters.

setPaymentMethodTokenizationResponse(paymentMethodTokenizationResponse)
Sets information from the gateway about the tokenized payment method.

setSalesforceResultCodeInfo(salesforceResultCodeInfo)
Sets the Salesforce-specific result code information. Payment gateways have many response codes for payment calls. Salesforce uses the result code information to map payment gateway codes to a predefined set of standard Salesforce result codes.

setAmount(amount)
Sets the transaction amount. Must be a non-negative value.

**Signature**

```global void setAmount(Double amount)```

**Parameters**

- **amount**: Type: **Double**
  The amount of the transaction.

**Return Value**

Type: **void**

setGatewayAvsCode(gatewayAvsCode)
Sets the AVS (address verification system) result code information that the gateway returned. Maximum length of 64 characters.

**Signature**

```global void setGatewayAvsCode(String gatewayAvsCode)```

**Parameters**

- **gatewayAvsCode**: Type: **String**
Used to verify the address mapped to a payment method when the payments platform requests tokenization from the payment gateway.

Return Value
Type: void

**setGatewayDate (gatewayDate)**
Sets the date that the sale occurred. Some gateways don’t send this value.

**Signature**

```plaintext
global void setGatewayDate(Datetime gatewayDate)
```

**Parameters**

- **gatewayDate**
  Type: Datetime

**Return Value**
Type: void

**setGatewayMessage (gatewayMessage)**
Sets error messages that the gateway returned for the sale request. Maximum length of 255 characters.

**Signature**

```plaintext
global void setGatewayMessage(String gatewayMessage)
```

**Parameters**

- **gatewayMessage**
  Type: String

**Return Value**
Type: void

**setGatewayReferenceDetails (gatewayReferenceDetails)**
Sets additional data that you can use for subsequent sales. You can use any data that isn’t normalized in financial entities. This field has a maximum length of 1000 characters and can store data as JSON or XML.

**Signature**

```plaintext
global void setGatewayReferenceDetails(String gatewayReferenceDetails)
```
Parameters

.gatewayReferenceDetails
Type: String

Return Value
Type: void

.setGatewayReferenceNumber(gatewayReferenceNumber)
Sets the unique gateway reference number for the transaction that the gateway returned. Maximum length of 255 characters.

Signature
global void setGatewayReferenceNumber(String gatewayReferenceNumber)

Parameters

gatewayReferenceNumber
Type: String
Unique authorization ID created by the payment gateway.

Return Value
Type: void

.setGatewayResultCode(gatewayResultCode)
Sets a gateway-specific result code. The code may be mapped to a Salesforce-specific result code. Maximum length of 64 characters.

Signature
global void setGatewayResultCode(String gatewayResultCode)

Parameters

gatewayResultCode
Type: String
Gateway-specific result code. Must be used to map a Salesforce-specific result code.

Return Value
Type: void

.setGatewayResultCodeDescription(gatewayResultCodeDescription)
Sets a description of the gateway-specific result code that a payment gateway returned. Maximum length of 1000 characters.
Signature

```java
global void setGatewayResultCodeDescription(String gatewayResultCodeDescription)
```

Parameters

gatewayResultCodeDescription
Type: String

Description of the gateway’s result code. Use this field to learn more about why the gateway returned a certain result code.

Return Value

Type: void

**setPaymentMethodTokenizationResponse (paymentMethodTokenizationResponse)**

Sets information from the gateway about the tokenized payment method.

Signature

```java
global void setPaymentMethodTokenizationResponse(commercepayments.PaymentMethodTokenizationResponse paymentMethodTokenizationResponse)
```

Parameters

paymentMethodTokenizationResponse
Type: PaymentMethodTokenizationResponse on page 342

Gateway response sent by payment gateway adapters for the payment method tokenization request. The response includes the payment method’s token ID value.

Return Value

Type: void

**setSalesforceResultCodeInfo(salesforceResultCodeInfo)**

Sets the Salesforce-specific result code information. Payment gateways have many response codes for payment calls. Salesforce uses the result code information to map payment gateway codes to a predefined set of standard Salesforce result codes.

Signature

```java
```

Parameters

salesforceResultCodeInfo
Type: SalesforceResultCodeInfo on page 377
Sets the Salesforce-specific result code information. Payment gateways have many response codes for payment calls. Salesforce uses the result code information to map payment gateway codes to a predefined set of standard Salesforce result codes.

Return Value

Type: void

SalesforceResultCode Enum

Defines the gateway call status values in Salesforce based on the call status values that the payment gateway returned.

Usage

Payment gateways can return many different responses. Salesforce maps these responses into one of seven possible Salesforce response values.

Enum Values

The following are the values of the `commercepayments.SalesforceResultCode` enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Decline</td>
<td>The gateway call failed, but it may still work if you try again. For example, the customer had insufficient funds or briefly lost their connection to the internet. This is also known as a &quot;soft decline.&quot;</td>
</tr>
<tr>
<td>Indeterminate</td>
<td>The gateway didn’t respond to the call and the user has to check the transaction request’s status. Indeterminate responses often occur following server timeouts, system failure, or any action that interrupts the gateway’s ability to process the payment.</td>
</tr>
<tr>
<td>PermanentFail</td>
<td>The customer’s bank recognized the payment account as closed, terminated, or fraudulent. The gateway won’t further calls from the payment method associate with the transaction. After a permanent fail response, the transaction changes its gateway status to Permanent Fail.</td>
</tr>
<tr>
<td>RequiresReview</td>
<td>The gateway call initially failed, but the payment method may still work after further evaluation. This response often happens when the customer bank requires more information about the payment request. In this case, the bank provides an authorization code manually when the payment manager calls the processor.</td>
</tr>
<tr>
<td>Success</td>
<td>The gateway processed the transaction successfully.</td>
</tr>
<tr>
<td>SystemError</td>
<td>Salesforce ended the payment request call before receiving a gateway response. System error responses often occur due to gateway server errors, invalid customer credentials, or anytime the request times out before receiving a gateway response. The failure occurs before the request reaches the gateway, so there’s no risk of an unaccounted payment remaining in the gateway. You can continue with the transaction by manually creating a payment.</td>
</tr>
<tr>
<td>ValidationErr</td>
<td>The gateway received incorrect customer payment information, such as misspelled credit card names or a CVV with missing numbers.</td>
</tr>
</tbody>
</table>
SalesforceResultCodeInfo

Stores Salesforce result code information from payment gateway adapters.

Namespace

CommercePayments on page 253

Usage

The constructor of this class takes no arguments. For example:

```java
```

Gateways can return the transaction result as either CustomMetadata or SalesforceResultCode.

IN THIS SECTION:

SalesforceResultCodeInfo Constructors

SalesforceResultCodeInfo Constructors

The following are constructors for SalesforceResultCodeInfo.

IN THIS SECTION:

SalesforceResultCodeInfo(customMetadataTypeInfo)

Constructor for providing the customMetadataTypeInfo for the result of the transaction.

SalesforceResultCodeInfo(salesforceResultCode)

Constructor that provides the salesforceResultCode for the transaction result.

SalesforceResultCodeInfo(customMetadataTypeInfo)

Constructor for providing the customMetadataTypeInfo for the result of the transaction.

Signature

```java
global SalesforceResultCodeInfo(commercepayments.CustomMetadataTypeInfo customMetadataTypeInfo)
```

Parameters

- `customMetadataTypeInfo`
  - Type: CommercePayments.CustomMetadataTypeInfo on page 321
  - Information about the metadata type.

SalesforceResultCodeInfo(salesforceResultCode)

Constructor that provides the salesforceResultCode for the transaction result.
Signature

global SalesforceResultCodeInfo(commercepayments.SalesforceResultCode
salesforceResultCode)

Parameters

salesforceResultCode

Type: SalesforceResultCode on page 376

The enum value for the result code.

ConnectApi Namespace

The ConnectApi namespace (also called Connect in Apex) provides classes for accessing the same data available in Connect REST API. Use Connect in Apex to create custom experiences in Salesforce.

For information about working with the ConnectApi classes, see Connect in Apex.

IN THIS SECTION:

ActionLinks Class
Create, delete, and get information about an action link group definition; get information about an action link group; get action link diagnostic information.

Announcements Class
Access information about announcements and post announcements.

BotVersionActivation Class
Access and update activation information of a bot version.

CdpCalculatedInsight Class
Create, delete, get, run, and update Data Cloud calculated insights.

CdpIdentityResolution Class
Create, delete, get, run, and update Data Cloud identity resolution rulesets.

CdpQuery Class
Get Data Cloud metadata and query data.

CdpSegment Class
Create, delete, get, publish, and update Data Cloud segments. Get segment members.

Chatter Class
Access information about followers and subscriptions for records.

ChatterFavorites Class
Chatter favorites give you easy access to topics, list views, and feed searches.

ChatterFeeds Class
Get, post, and delete feed elements, likes, comments, and bookmarks. You can also search feed elements, share feed elements, and vote on polls.

ChatterGroups Class
Information about groups, such as the group’s members, photo, and the groups the specified user is a member of. Add members to a group, remove members, and change the group photo.
ChatterMessages Class
Get, send, search, and reply to private messages. You can also get and search private conversations, mark conversations as read, and get a count of unread private messages.

ChatterUsers Class
Access information about users, such as activity, followers, subscriptions, files, and groups.

Clm Class
Create and update Contract Lifecycle Management (CLM) contracts using object ID.

CommerceBuyerExperience Class
Create, delete, or get commerce addresses. Get order delivery group, order item, order shipments, shipment items, and order summaries. Get adjustments for order items and order summaries.

CommerceCart Class
Get, create, update, and delete carts. Get cart items, add items to carts, update and delete cart items.

CommerceCatalog Class
Get products, product categories, and product category paths.

CommercePromotions Class
Evaluate promotions for Commerce orders. Get coupon code redemption usage.

CommerceSearch Class

CommerceSearchConnectFamily Class
Search products by search term or category in a webstore.

CommerceSearchSettings Class
Get indexes. Get index logs. Create an index of a product catalog.

CommerceStorePricing Class
Get product prices.

CommerceWishlist Class
Get, create, update, and delete wishlists. Add wishlists to carts. Get wishlist items, add items to wishlists, and delete wishlist items.

Communities Class
Get information about Experience Cloud sites in your org.

CommunityModeration Class
Get information about flagged feed items and comments in an Experience Cloud site. Add and remove flags from comments and feed items.

ContentHub Class
Access Files Connect repositories and their files and folders.

ConversationApplicationDefinition Class
Access information about a conversation application definition.

Datacloud Class
Purchase Data.com contact or company records, and retrieve purchase information.

EmailMergeFieldService Class
Extract a list of merge fields for an object. A merge field is a field you can put in an email template, mail merge template, custom link, or formula to incorporate values from a record.
**EmployeeProfiles Class**
Get, set and crop, and delete employee banner photos and photos.

**ExternalEmailServices Class**
Access information about integration with external email services, such as sending email within Salesforce through an external email account.

**ExternalManagedAccount Class**
Get externally managed accounts.

**FieldService Class**
Preview and create shifts from a pattern or filter fields on recordset filter criteria.

**FulfillmentOrder Class**
Fulfill orders in Order Management.

**Knowledge Class**
Get information about trending articles in Experience Cloud sites.

**LightningScheduler Class**
Create and update service appointments.

**ManagedContent Class**
Get managed content versions. Get a managed content space.

**ManagedContentDelivery Class**
Get collection items. Get a managed content channel. Get managed content.

**ManagedTopics Class**
Get managed topics in an Experience Cloud site. Create, delete, and reorder managed topics.

**MarketingIntegration Class**
Get, save, and submit a microsites marketing integration form for an Experience Cloud site.

**Mentions Class**
Access information about mentions. A mention is an "@" character followed by a user or group name. When a user or group is mentioned, they receive a notification.

**Missions Class**
Export and purge mission activity for users. Get a user's mission progress. Update mission activity counts for users.

**NamedCredentials Class**
Create, refresh, get, delete, replace, and update credentials. Create and get external credentials. Create and get named credentials. Get the URL for the OAuth token flow for an external credential.

**NavigationMenu Class**
Get navigation menu items for an Experience Cloud site.

**NextBestAction Class**
Execute recommendation strategies, get recommendations, manage recommendation reactions.

**OmnichannelInventoryService Class**
Route orders to inventory locations in Order Management.

**Orchestration Class**
Get orchestration instances.

**OrderPaymentSummary Class**
Work with payments in Order Management.
OrderSummary Class
Work with orders in Order Management.

OrderSummaryCreation Class
Create Order Summaries in Order Management.

Organization Class
Access information about an org.

PardotBusinessUnitContext Class
Get the Pardot business units the context user has access to.

Payments Class
Authorize a payment, capture an authorized payment, and refund an authorized payment.

Personalization Class
Get assigned personalization audiences that match the user context. Create, get, update, and delete an audience. Get personalization targets that match the user context, based on the assigned audiences that include the user. Create and update targets. Get and delete a target.

PickTicket Class
Create tickets to fulfill orders.

QuestionAndAnswers Class
Access question and answers suggestions.

Recommendations Class
Get and reject Chatter, custom, and static recommendations. Create, get, update, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

Records Class
Access information about record motifs, which are small icons used to distinguish record types in the Salesforce UI.

Repricing Class
Perform functions related to repricing orders in Order Management.

ReturnOrder Class
Process ReturnOrders in Order Management.

Routing Class
Route orders to inventory locations in Order Management.

SalesforceInbox Class
Access information about Automated Activity Capture, which is available in Einstein and Salesforce Inbox.

Sites Class
Search an Experience Cloud site.

SmartDataDiscovery Class
Get predictions on Salesforce objects.

SocialEngagement Class
Manage information about social accounts or fan pages for social networks.

Surveys Class
Send survey invitations by email.

TaxPlatform Class
Apply or cancel tax.
Topics Class
Access information about topics, such as their descriptions, the number of people talking about them, related topics, and information about groups contributing to the topic. Update a topic’s name or description, merge topics, and add and remove topics from records and feed items.

UserProfiles Class
Access user profile data. The user profile data populates the profile page (also called the Chatter profile page). This data includes user information (such as address, manager, and phone number), some user capabilities (permissions), and a set of subtab apps, which are custom tabs on the profile page.

Zones Class
Access information about Chatter Answers zones in your organization. Zones organize questions into logical groups, with each zone having its own focus and unique questions.

ConnectApi Input Classes
Some ConnectApi methods take arguments that are instances of ConnectApi input classes.

ConnectApi Output Classes
Most ConnectApi methods return instances of ConnectApi output classes.

ConnectApi Enums
 Enums specific to the ConnectApi namespace.

ConnectApi Exceptions
The ConnectApi namespace contains exception classes.

ConnectApi Utilities
The ConnectApi namespace contains a utility class.

ConnectApi Release Notes
Use the Salesforce Release Notes to learn about the most recent updates and changes to the ConnectApi namespace in Apex.

ActionLinks Class
Create, delete, and get information about an action link group definition; get information about an action link group; get action link diagnostic information.

Namespace
ConnectApi

Usage
An action link is a button on a feed element. Clicking an action link can take a user to a Web page, initiate a file download, or invoke an API call to Salesforce or to an external server. An action link includes a URL and an HTTP method, and can include a request body and header information, such as an OAuth token for authentication. Use action links to integrate Salesforce and third-party services into the feed so that users can drive productivity and accelerate innovation.

There are two views of an action link and an action link group: the definition, and the context user’s view. The definition includes potentially sensitive information, such as authentication information. The context user’s view is filtered by visibility options and the values reflect the state of the context user.

Action link definition can be sensitive to a third party (for example, OAuth bearer token headers). For this reason, only calls made from the Apex namespace that created the action link definition can read, modify, or delete the definition. In addition, the user making the
call must have created the definition or have View All Data permission. Use these methods to operate on action link group definitions (which contain action link definitions).

- `createActionLinkGroupDefinition(communityId, actionLinkGroup)`
- `deleteActionLinkGroupDefinition(communityId, actionLinkGroupId)`
- `getActionLinkGroupDefinition(communityId, actionLinkGroupId)`

Use these methods to operate on a context user’s view of an action link or an action link group.

- `getActionLink(communityId, actionLinkId)`
- `getActionLinkGroup(communityId, actionLinkGroupId)`
- `getActionLinkDiagnosticInfo(communityId, actionLinkId)`

For information about how to use action links, see Working with Action Links.

**ActionLinks Methods**

These are methods for ActionLinks. All methods are static.

**IN THIS SECTION:**

- `createActionLinkGroupDefinition(communityId, actionLinkGroup)`
  Create an action link group definition. To associate an action link group with a feed element, first create an action link group definition. Then post a feed element with an associated actions capability.
- `deleteActionLinkGroupDefinition(communityId, actionLinkGroupId)`
  Delete an action link group definition. Deleting an action link group definition removes all references to it from feed elements.
- `getActionLink(communityId, actionLinkId)`
  Get information about an action link, including state for the context user.
- `getActionLinkGroup(communityId, actionLinkGroupId)`
  Get information about an action link group including state for the context user.
- `getActionLinkDiagnosticInfo(communityId, actionLinkId)`
  Get diagnostic information returned when an action link executes. Diagnostic information is given only for users who can access the action link.
- `getActionLinkGroupDefinition(communityId, actionLinkGroupId)`
  Get information about an action link group definition.

**createActionLinkGroupDefinition(communityId, actionLinkGroup)**

Create an action link group definition. To associate an action link group with a feed element, first create an action link group definition. Then post a feed element with an associated actions capability.

**API Version**

33.0

**Requires Chatter**

No
Signature

```java
public static ConnectApi.ActionLinkGroupDefinition createActionLinkGroupDefinition(String communityId, ConnectApi.ActionLinkGroupDefinitionInput actionLinkGroup)
```

Parameters

`communityId`
Type: String
ID for an Experience Cloud site, internal, or null.

`actionLinkGroup`
Type: `ConnectApi.ActionLinkGroupDefinitionInput`
A `ConnectApi.ActionLinkGroupDefinitionInput` object that defines the action link group.

Return Value

Type: `ConnectApi.ActionLinkGroupDefinition`

Usage

An action link is a button on a feed element. Clicking an action link can take a user to a Web page, initiate a file download, or invoke an API call to Salesforce or to an external server. An action link includes a URL and an HTTP method, and can include a request body and header information, such as an OAuth token for authentication. Use action links to integrate Salesforce and third-party services into the feed so that users can drive productivity and accelerate innovation.

All action links must belong to a group. Action links in a group are mutually exclusive and share some properties. Define standalone actions in their own action group.

Information in the action link group definition can be sensitive to a third party (for example, OAuth bearer token headers). For this reason, only calls made from the Apex namespace that created the action link group definition can read, modify, or delete the definition. In addition, the user making the call must have created the definition or have View All Data permission.

**Note:** Invoking `ApiAsync` action links from an app requires a call to set the status. However, there isn’t currently a way to set the status of an action link using Apex. To set the status, use Connect REST API. See the Action Link resource in the Connect REST API Developer Guide for more information.

Example for Defining an Action Link and Posting with a Feed Element

For more information about this example, see Define an Action Link and Post with a Feed Element.

```java
ConnectApi.ActionLinkGroupDefinitionInput actionLinkGroupDefinitionInput = new ConnectApi.ActionLinkGroupDefinitionInput();
ConnectApi.ActionLinkDefinitionInput actionLinkDefinitionInput = new ConnectApi.ActionLinkDefinitionInput();
ConnectApi.RequestHeaderInput requestHeaderInput1 = new ConnectApi.RequestHeaderInput();
ConnectApi.RequestHeaderInput requestHeaderInput2 = new ConnectApi.RequestHeaderInput();

// Create the action link group definition.
actionLinkGroupDefinitionInput.actionLinks = New List<ConnectApi.ActionLinkDefinitionInput>();
actionLinkGroupDefinitionInput.executionsAllowed = ConnectApi.ActionLinkExecutionsAllowed.OncePerUser;
actionLinkGroupDefinitionInput.category = ConnectApi.PlatformActionGroupCategory.Primary;
```
To Do: Verify that the date is in the future.
To Do: Substitute an OAuth value for your Salesforce org.

```
datetime myDate = datetime.newInstance(2016, 3, 1);
actionLinkGroupDefinitionInput.expirationDate = myDate;

// Create the action link definition.
actionLinkDefinitionInput.actionType = ConnectApi.ActionLinkType.Api;
actionLinkDefinitionInput.actionUrl = '/services/data/v33.0/chatter/feed-elements';
actionLinkDefinitionInput.headers = new List<ConnectApi.RequestHeaderInput>();
actionLinkDefinitionInput.labelKey = 'Post';
actionLinkDefinitionInput.method = ConnectApi.HttpRequestMethod.HttpPost;
actionLinkDefinitionInput.requestBody = '{"subjectId": "me","feedElementType": "FeedItem","body": [{"type": "Text","text": "This is a test post created via an API action link."}]};
actionLinkDefinitionInput.requiresConfirmation = true;

requestHeaderInput1.name = 'Authorization';
requestHeaderInput1.value = 'OAuth 00DD00000007WNP!ARsAQwveV0zzAV847FTT4zF.85w.EwsPbUgXa4Sajsp';
actionLinkDefinitionInput.headers.add(requestHeaderInput1);

requestHeaderInput2.name = 'Content-Type';
requestHeaderInput2.value = 'application/json';
actionLinkDefinitionInput.headers.add(requestHeaderInput2);

// Add the action link definition to the action link group definition.
actionLinkGroupDefinitionInput.actionLinks.add(actionLinkDefinitionInput);

// Instantiate the action link group definition.
ConnectApi.ActionLinkGroupDefinition actionLinkGroupDefinition =
    ConnectApi.ActionLinks.createActionLinkGroupDefinition(Network.getNetworkId(),
    actionLinkGroupDefinitionInput);

ConnectApi.FeedItemInput feedItemInput = new ConnectApi.FeedItemInput();
ConnectApi.FeedElementCapabilitiesInput feedElementCapabilitiesInput = new
    ConnectApi.FeedElementCapabilitiesInput();
ConnectApi.AssociatedActionsCapabilityInput associatedActionsCapabilityInput = new
    ConnectApi.AssociatedActionsCapabilityInput();
ConnectApi.TextSegmentInput textSegmentInput = new ConnectApi.TextSegmentInput();

// Set the properties of the feedItemInput object.
feedItemInput.body = messageBodyInput;
feedItemInput.capabilities = feedElementCapabilitiesInput;
feedItemInput.subjectId = 'me';

// Create the text for the post.
messageBodyInput.messageSegments = new List<ConnectApi.MessageSegmentInput>();
textSegmentInput.text = 'Click to post a feed item.';
messageBodyInput.messageSegments.add(textSegmentInput);

// The feedElementCapabilitiesInput object holds the capabilities of the feed item.
```
// Define an associated actions capability to hold the action link group.
// The action link group ID is returned from the call to create the action link group
definition.
feedElementCapabilitiesInput.associatedActions = associatedActionsCapabilityInput;
associatedActionsCapabilityInput.actionLinkGroupIds = new List<String>();
associatedActionsCapabilityInput.actionLinkGroupIds.add(actionLinkGroupDefinition.id);

// Post the feed item.
ConnectApi.FeedElement feedElement =
ConnectApi.ChatterFeeds.postFeedElement(Network.getNetworkId(), feedItemInput);

Example for Defining an Action Link in a Template and Posting with a Feed Element

For more information about this example, see Define an Action Link in a Template and Post with a Feed Element.

// Get the action link group template Id.
ActionLinkGroupTemplate template = [SELECT Id FROM ActionLinkGroupTemplate WHERE
DeveloperName='Doc_Example'];

// Add binding name-value pairs to a map.
// The names are defined in the action link template(s) associated with the action link
group template.
// Get them from Setup UI or SOQL.
Map<String, String> bindingMap = new Map<String, String>();
bindingMap.put('ApiVersion', 'v33.0');
bindingMap.put('Text', 'This post was created by an API action link.');
bindingMap.put('SubjectId', 'me');

// Create ActionLinkTemplateBindingInput objects from the map elements.
List<ConnectApi.ActionLinkTemplateBindingInput> bindingInputs = new
List<ConnectApi.ActionLinkTemplateBindingInput>();
for (String key : bindingMap.keySet()) {
    ConnectApi.ActionLinkTemplateBindingInput bindingInput = new
ConnectApi.ActionLinkTemplateBindingInput();
    bindingInput.key = key;
    bindingInput.value = bindingMap.get(key);
    bindingInputs.add(bindingInput);
}

// Set the template Id and template binding values in the action link group definition.
ConnectApi.ActionLinkGroupDefinitionInput actionLinkGroupDefinitionInput = new
ConnectApi.ActionLinkGroupDefinitionInput();
actionLinkGroupDefinitionInput.templateId = template.id;
actionLinkGroupDefinitionInput.templateBindings = bindingInputs;

// Instantiate the action link group definition.
ConnectApi.ActionLinkGroupDefinition actionLinkGroupDefinition =
    ConnectApi.ActionLinks.createActionLinkGroupDefinition(Network.getNetworkId(),
actionLinkGroupDefinitionInput);

ConnectApi.FeedItemInput feedItemInput = new ConnectApi.FeedItemInput();
ConnectApi.FeedElementCapabilitiesInput feedElementCapabilitiesInput = new
ConnectApi.FeedElementCapabilitiesInput();
ConnectApi.AssociatedActionsCapabilityInput associatedActionsCapabilityInput = new ConnectApi.AssociatedActionsCapabilityInput();
ConnectApi.TextSegmentInput textSegmentInput = new ConnectApi.TextSegmentInput();

// Define the FeedItemInput object to pass to postFeedElement
feedItemInput.body = messageBodyInput;
feedItemInput.capabilities = feedElementCapabilitiesInput;
feedItemInput.subjectId = 'me';

// The MessageBodyInput object holds the text in the post
messageBodyInput.messageSegments = new List<ConnectApi.MessageSegmentInput>();
textSegmentInput.text = 'Click to post a feed item.';
messageBodyInput.messageSegments.add(textSegmentInput);

// The FeedElementCapabilitiesInput object holds the capabilities of the feed item.
// For this feed item, we define an associated actions capability to hold the action link group.
// The action link group ID is returned from the call to create the action link group definition.
feedElementCapabilitiesInput.associatedActions = associatedActionsCapabilityInput;
associatedActionsCapabilityInput.actionLinkGroupIds = new List<String>();
associatedActionsCapabilityInput.actionLinkGroupIds.add(actionLinkGroupDefinition.id);

// Post the feed item.
ConnectApi.FeedElement feedElement =
ConnectApi.ChatterFeeds.postFeedElement(Network.getNetworkId(), feedItemInput);

deleteActionLinkGroupDefinition(communityId, actionLinkGroupId)

Delete an action link group definition. Deleting an action link group definition removes all references to it from feed elements.

API Version
33.0

Requires Chatter
No

Signature

public static void deleteActionLinkGroupDefinition(String communityId, String actionLinkGroupId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.
**actionLinkId**
  Type: `String`
  The ID of the action link.

**Return Value**
Type: Void

**Usage**
Information in the action link group definition can be sensitive to a third party (for example, OAuth bearer token headers). For this reason, only calls made from the Apex namespace that created the action link group definition can read, modify, or delete the definition. In addition, the user making the call must have created the definition or have View All Data permission.

**getActionLink(communityId, actionLinkId)**
Get information about an action link, including state for the context user.

**API Version**
33.0

**Requires Chatter**
No

**Signature**
```java
public static ConnectApi.PlatformAction getActionLink(String communityId, String actionLinkId)
```

**Parameters**
- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.
- `actionLinkId`
  Type: `String`
  The ID of the action link.

**Return Value**
Type: `ConnectApi.PlatformAction`

**getActionLinkDiagnosticInfo(communityId, actionLinkId)**
Get diagnostic information returned when an action link executes. Diagnostic information is given only for users who can access the action link.
API Version
33.0

Requires Chatter
No

Signature
public static ConnectApi.ActionLinkDiagnosticInfo getActionLinkDiagnosticInfo(String communityId, String actionLinkId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

actionLinkId
Type: String
The ID of the action link.

Return Value
Type: ConnectApi.ActionLinkDiagnosticInfo

getActionLinkGroup(communityId, actionLinkGroupId)
Get information about an action link group including state for the context user.

API Version
33.0

Requires Chatter
No

Signature
public static ConnectApi.PlatformActionGroup getActionLinkGroup(String communityId, String actionLinkGroupId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.
**actionLinkGroupId**
Type: **String**
The ID of the action link group.

**Return Value**
Type: **ConnectApi.PlatformActionGroup**

**Usage**
All action links must belong to a group. Action links in a group are mutually exclusive and share some properties. Action link groups are accessible by clients, unlike action link group definitions.

**getActionLinkGroupDefinition(communityId, actionLinkGroupId)**
Get information about an action link group definition.

**API Version**
33.0

**Requires Chatter**
No

**Signature**
```java
public static ConnectApi.ActionLinkGroupDefinition getActionLinkGroupDefinition(String communityId, String actionLinkGroupId)
```

**Parameters**
- **communityId**
  Type: **String**
  ID for an Experience Cloud site, internal, or null.
- **actionLinkGroupId**
  Type: **String**
  The ID of the action link group.

**Return Value**
Type: **ConnectApi.ActionLinkGroupDefinition**

**Usage**
Information in the action link group definition can be sensitive to a third party (for example, OAuth bearer token headers). For this reason, only calls made from the Apex namespace that created the action link group definition can read, modify, or delete the definition. In addition, the user making the call must have created the definition or have View All Data permission.
Announcements Class

Access information about announcements and post announcements.

Namespace

ConnectApi

Usage

Use the ConnectApi.Announcements class to get, create, update, and delete announcements. Use an announcement to highlight information. Users can discuss, like, and post comments on announcements. Deleting the feed post deletes the announcement.

This image shows an announcement displayed in a group. Creating an announcement also creates a feed item with the announcement text.

An announcement displays in a designated location in the Salesforce UI until 11:59 p.m. on its expiration date, unless it's deleted or replaced by another announcement.

Announcements Methods

The following are methods for Announcements. All methods are static.

IN THIS SECTION:

- deleteAnnouncement(communityId, announcementId)
  Delete an announcement.
- getAnnouncement(communityId, announcementId)
  Get an announcement.
**getAnnouncements(communityId, parentId)**
Get the first page of announcements.

**getAnnouncements(communityId, parentId, pageParam, pageSize)**
Get a page of announcements.

**postAnnouncement(communityId, announcement)**
Post an announcement.

**updateAnnouncement(communityId, announcementId, expirationDate)**
Update the expiration date of an announcement.

**deleteAnnouncement(communityId, announcementId)**
Delete an announcement.

**API Version**
31.0

**Requires Chatter**
Yes

**Signature**
```java
public static void deleteAnnouncement(String communityId, String announcementId)
```

**Parameters**
- **communityId**
  Type: String
  ID for an Experience Cloud site, internal, or null.
- **announcementId**
  Type: String
  An announcement ID, which has a prefix of 0BT.

**Return Value**
Type: Void

**Usage**
To get a list of announcements in a group, call `getAnnouncements(communityId, parentId)` or `getAnnouncements(communityId, parentId, pageParam, pageSize)`.
To post an announcement to a group, call `postAnnouncement(communityId, announcement)`.

**getAnnouncement(communityId, announcementId)**
Get an announcement.
API Version
31.0

Requires Chatter
Yes

Signature
public static ConnectApi.Announcement getAnnouncement(String communityId, String announcementId)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.
announcementId
Type: String
An announcement ID, which has a prefix of 0BT.

Return Value
Type: ConnectApi.Announcement

Usage
To get a list of announcements in a group, call getAnnouncements(communityId, parentId) or getAnnouncements(communityId, parentId, pageParam, pageSize).
To post an announcement to a group, call postAnnouncement(communityId, announcement).

getAnnouncements(communityId, parentId)
Get the first page of announcements.

API Version
36.0

Available to Guest Users
38.0

Requires Chatter
Yes
public static ConnectApi.AnnouncementPage getAnnouncements(String communityId, String parentId)

Parameters

communityId  
Type: String  
Id for an Experience Cloud site, internal, or null.

parentId  
Type: String  
Id of the parent entity for the announcement, that is, a group ID when the announcement appears in a group.

Return Value

Type: ConnectApi.AnnouncementPage

getAnnouncements(communityId, parentId, pageParam, pageSize)
Get a page of announcements.

API Version

36.0

Available to Guest Users

38.0

Requires Chatter

Yes

Signature

public static ConnectApi.AnnouncementPage getAnnouncements(String communityId, String parentId, Integer pageParam, Integer pageSize)

Parameters

communityId  
Type: String  
Id for an Experience Cloud site, internal, or null.

parentId  
Type: String  
Id of the parent entity for the announcement, that is, a group ID when the announcement appears in a group.

pageParam  
Type: Integer

pageSize  
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

`pageSize`
Type: `Integer`
Specifies the number of announcements per page.

**Return Value**
Type: `ConnectApi.AnnouncementPage`

**postAnnouncement(communityId, announcement)**
Post an announcement.

**API Version**
36.0

**Requires Chatter**
Yes

**Signature**

```
public static ConnectApi.Announcement postAnnouncement(String communityId,
ConnectApi.AnnouncementInput announcement)
```

**Parameters**

`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

`announcement`
Type: `ConnectApi.AnnouncementInput`
A `ConnectApi.AnnouncementInput` object.

**Return Value**
Type: `ConnectApi.Announcement` on page 1689

**updateAnnouncement(communityId, announcementId, expirationDate)**
Update the expiration date of an announcement.

**API Version**
31.0
Requires Chatter
Yes

Signature

public static ConnectApi.Announcement updateAnnouncement(String communityId, String announcementId, Datetime expirationDate)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

announcementId
Type: String
An announcement ID, which has a prefix of 0BT.

expirationDate
Type: Datetime
The Salesforce UI displays an announcement until 11:59 p.m. on this date unless another announcement is posted first. The Salesforce UI ignores the time value in the expirationDate. However, you can use the time value to create your own display logic in your own UI.

Return Value
Type: ConnectApi.Announcement

Usage
To get a list of announcements in a group, call getAnnouncements(communityId, parentId) or getAnnouncements(communityId, parentId, pageParam, pageSize).
To post an announcement to a group, call postAnnouncement(communityId, announcement).

BotVersionActivation Class
Access and update activation information of a bot version.

Namespace
ConnectApi

BotVersionActivation Methods
The following are methods for BotVersionActivation. All methods are static.
IN THIS SECTION:

getVersionActivationInfo(botVersionId)
Get the active or inactive status of the bot version.

updateVersionStatus(botVersionId, status, postBody)
Update the status of the specified bot version.

getVersionActivationInfo (botVersionId)
Get the active or inactive status of the bot version.

API Version
50.0

Requires Chatter
No

Signature
public static ConnectApi.BotVersionActivationInfo getVersionActivationInfo(String botVersionId)

Parameters
botVersionId
Type: String
ID of the bot version.

Return Value
Type: ConnectApi.BotVersionActivationInfo

Usage
To access this method, enable the bot feature, and the user must be an admin or have the Manage Bots or Manage Bots Training Data user permissions.

updateVersionStatus(botVersionId, status, postBody)
Update the status of the specified bot version.

API Version
50.0

Requires Chatter
No
**Signature**

```java
ConnectApi.BotVersionActivationInput postBody)
```

**Parameters**

- **botVersionId**
  - Type: String
  - ID of the bot version.

- **status**
  - Type: `ConnectApi.BotVersionActivationStatus`
  - Activation status of the bot version. Values are:
    - Active
    - Inactive
  - Activation status must be specified in the `status` or `postBody` parameter.

- **postBody**
  - Type: `ConnectApi.BotVersionActivationInput`
  - Parameters to update for the bot version. Activation status must be specified in the `status` or `postBody` parameter.

**Return Value**

Type: `ConnectApi.BotVersionActivationInfo`

**Usage**

To access this method, enable the bot feature, and the user must be an admin or have the Manage Bots or Manage Bots Training Data user permissions.

**CdpCalculatedInsight Class**

Create, delete, get, run, and update Data Cloud calculated insights.

**Namespace**

`ConnectApi`

**CdpCalculatedInsight Methods**

The following are methods for `CdpCalculatedInsight`. All methods are static.

**IN THIS SECTION:**

- `createCalculatedInsight(input)`
  - Create a calculated insight.
deleteCalculatedInsight(apiName)
Delete a calculated insight.

getCalculatedInsight(apiName)
Get a calculated insight.

getCalculatedInsights(definitionType, batchSize, offset, orderby, dataspace)
Get calculated insights.

getCalculatedInsights(definitionType, batchSize, offset, orderby, dataspace, pageToken)
Get a page of calculated insights.

runCalculatedInsight(apiName)
Run a calculated insight.

updateCalculatedInsight(apiName, input)
Update a calculated insight.

createCalculatedInsight(input)
Create a calculated insight.

API Version

57.0

Requires Chatter

No

Signature

public static ConnectApi.CdpCalculatedInsightOutput
createCalculatedInsight(ConnectApi.CdpCalculatedInsightInput input)

Parameters

input
  Type: ConnectApi.CdpCalculatedInsightInput
  Input representation for a calculated insight.

Return Value

Type: ConnectApi.CdpCalculatedInsightOutput

deleteCalculatedInsight(apiName)
Delete a calculated insight.

API Version

57.0
Requires Chatter
No

Signature

```java
public static Void deleteCalculatedInsight(String apiName)
```

Parameters

```java
apiName
    Type: String
    API name of the calculated insight to delete.
```

Return Value

Type: Void

`getCalculatedInsight(apiName)`
Get a calculated insight.

API Version

57.0

Requires Chatter
No

Signature

```java
public static ConnectApi.CdpCalculatedInsightOutput getCalculatedInsight(String apiName)
```

Parameters

```java
apiName
    Type: String
    API name of the calculated insight to get.
```

Return Value

Type: `ConnectApi.CdpCalculatedInsightOutput`

`getCalculatedInsights(definitionType, batchSize, offset, orderby, dataspace)`
Get calculated insights.

API Version

56.0
Requires Chatter
No

Signature

```java
public static ConnectApi.CdpCalculatedInsightPage getCalculatedInsights(String definitionType, Integer batchSize, Integer offset, String orderby, String dataspace)
```

Parameters

- **definitionType**
  - Type: `String`
  - Definition type of the calculated insight. Values are:
    - `CalculatedMetric`
    - `ExternalMetric`
    - `StreamingMetric`

- **batchSize**
  - Type: `Integer`
  - Number of items to return. Values are from 1–300. If unspecified, the default value is 25.

- **offset**
  - Type: `Integer`
  - Number of rows to skip before returning results. If unspecified, no rows are skipped.

- **orderby**
  - Type: `String`
  - Sort order for the result set, such as `GenderId__c ASC, Occupation__c DESC`. If unspecified, items are returned in the order they are retrieved.

- **dataspace**
  - Type: `String`
  - Name of the data space.

Return Value

Type: `ConnectApi.CdpCalculatedInsightPage`

```java
getCalculatedInsights(definitionType, batchSize, offset, orderby, dataspace, pageToken)
```

Get a page of calculated insights.

API Version

57.0

Requires Chatter
No
Signature

public static ConnectApi.CdpCalculatedInsightPage getCalculatedInsights(String definitionType, Integer batchSize, Integer offset, String orderby, String dataspace, String pageToken)

Parameters

definitionType
Type: String
Definition type of the calculated insight. Values are:
- CalculatedMetric
- ExternalMetric
- StreamingMetric

batchSize
Type: Integer
Number of items to return. Values are from 1–300. If unspecified, the default value is 25.

offset
Type: Integer
Number of rows to skip before returning results. If unspecified, no rows are skipped.

orderby
Type: String
Sort order for the result set, such as GenderId__c ASC, Occupation__c DESC. If unspecified, items are returned in the order they are retrieved.

dataspace
Type: String
Name of the data space.

pageToken
Type: String
Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in null, the first page is returned.

Return Value
Type: ConnectApi.CdpCalculatedInsightPage

runCalculatedInsight(apiName)
Run a calculated insight.

API Version
57.0
Requires Chatter
No

Signature
public static ConnectApi.CdpCalculatedInsightStandardActionResponseRepresentation runCalculatedInsight(String apiName)

Parameters
apiName
  Type: String
  API name of the calculated insight to run.

Return Value
Type: ConnectApi.CdpCalculatedInsightStandardActionResponseRepresentation

updateCalculatedInsight(apiName, input)
Update a calculated insight.

API Version
57.0

Requires Chatter
No

Signature
public static ConnectApi.CdpCalculatedInsightOutput updateCalculatedInsight(String apiName, ConnectApi.CdpCalculatedInsightInput input)

Parameters
apiName
  Type: String
  API name of the calculated insight to update.

input
  Type: ConnectApi.CdpCalculatedInsightInput
  Input representation for a calculated insight.

Return Value
Type: ConnectApi.CdpCalculatedInsightOutput
CdpIdentityResolution Class
Create, delete, get, run, and update Data Cloud identity resolution rulesets.

Namespace
ConnectApi

CdpIdentityResolution Methods
The following are methods for CdpIdentityResolution. All methods are static.

IN THIS SECTION:
createIdentityResolution(input)
Create an identity resolution ruleset.
deleteIdentityResolution(identityResolution)
Delete an identity resolution ruleset.
getIdentityResolution(identityResolution)
Get an identity resolution ruleset.
getIdentityResolutions()
Get identity resolution rulesets.
runIdentityResolutionNow(identityResolution, input)
Trigger an immediate identity resolution ruleset job run.
updateIdentityResolution(identityResolution, input)
Update an identity resolution ruleset.

createIdentityResolution(input)
Create an identity resolution ruleset.

API Version
S7.0

Requires Chatter
No

Signature
public static ConnectApi.CdpIdentityResolutionOutput createIdentityResolution(ConnectApi.CdpIdentityResolutionConfigInput input)

Parameters
input
Type: ConnectApi.CdpIdentityResolutionConfigInput
Input representation for creating an identity resolution ruleset.

Return Value
Type: ConnectApi.CdpIdentityResolutionOutput

deleIdentityResolution(identityResolution)
Delete an identity resolution ruleset.

API Version
57.0

Requires Chatter
No

Signature
public static Void deleteIdentityResolution(String identityResolution)

Parameters

identityResolution
Type: String
Developer name or ID of the ruleset.

Return Value
Type: Void

getIdentityResolution(identityResolution)
Get an identity resolution ruleset.

API Version
57.0

Requires Chatter
No

Signature
public static ConnectApi.CdpIdentityResolutionOutput getIdentityResolution(String identityResolution)
Parameters

\( \text{identityResolution} \)
Type: \text{String}
Developer name or ID of the ruleset.

Return Value
Type: \text{ConnectApi.CdpIdentityResolutionOutput}

\text{getIdentityResolutions()}\)
Get identity resolution rulesets.

API Version
57.0

Requires Chatter
No

Signature
\text{public static ConnectApi.CdpIdentityResolutionsOutput getIdentityResolutions()}

Return Value
Type: \text{ConnectApi.CdpIdentityResolutionsOutput}

\text{runIdentityResolutionNow(identityResolution, input)}\)
Trigger an immediate identity resolution ruleset job run.

API Version
57.0

Requires Chatter
No

Signature
\text{public static ConnectApi.CdpIdentityResolutionRunNowOutput runIdentityResolutionNow(String identityResolution,}
\text{ConnectApi.CdpIdentityResolutionRunNowInput input)}
Parameters

identityResolution
  Type: String
  Developer name of the ruleset.

input
  Type: ConnectApi.CdpIdentityResolutionRunNowInput
  Input representation for running an identity resolution ruleset job on demand.

Return Value

Type: ConnectApi.CdpIdentityResolutionRunNowOutput

updateIdentityResolution(identityResolution, input)

Update an identity resolution ruleset.

API Version

57.0

Requires Chatter

No

Signature


Parameters

identityResolution
  Type: String
  Developer name or ID of the ruleset.

input
  Type: ConnectApi.CdpIdentityResolutionConfigPatchInput
  Input representation for updating an identity resolution ruleset.

Return Value

Type: ConnectApi.CdpIdentityResolutionOutput

CdpQuery Class

Get Data Cloud metadata and query data.
Namespace

ConnectApi

CdpQuery Methods

The following are methods for CdpQuery. All methods are static.

IN THIS SECTION:

- **getAllMetadata()**: Get all metadata, including Calculated Insights, Engagement, Profile, and other objects, as well as their relationships to other objects.
- **getAllMetadata(entityType, entityCategory, entityName)**: Get all metadata, filtering for entity type, category, and name.
- **getInsightsMetadata()**: Get Insight metadata, including Calculated Insight objects, their dimensions and measures.
- **getInsightsMetadata(ciName)**: Get metadata for a Calculated Insight object. Metadata includes dimensions and measures.
- **getProfileMetadata()**: Get metadata for data model objects in the profile category, including Individual, Contact Point Email, Unified Individual, and Contact Point Address objects. Metadata includes the objects, their fields, and category.
- **getProfileMetadata(dataModelName)**: Get metadata for a data model object in the profile category, such as Individual, Contact Point Email, Unified Individual, and Contact Point Address. Metadata includes the list of fields, data types, and indexes available for lookup.
- **nextBatchAnsiSqlV2(nextBatchId)**: Get the next batch of data across data model, lake, unified, and linked objects.
- **queryANSISql(input)**: Synchronously query data across data model, lake, unified, and linked objects. This query returns up to 4,999 rows.
- **queryANSISql(input, batchSize, offset, orderby)**: Synchronously query data across data model, lake, unified, and linked objects. Specify batch size, offset, and order of the results. This query returns up to 4,999 rows.
- **queryAnsiSqlV2(input)**: Query up to 8 MB of data across data model, lake, unified, and linked objects.
- **queryCalculatedInsights(ciName, dimensions, measures, orderby, filters, batchSize, offset)**: Query a Calculated Insight object.
- **queryCalculatedInsights(ciName, dimensions, measures, orderby, filters, batchSize, offset, timeGranularity)**: Query a Calculated Insight object within a specified time range.
- **queryProfileApi(dataModelName, filters, fields, batchSize, offset, orderby)**: Query a Profile data model object using filters.
- **queryProfileApi(dataModelName, id, searchKey, filters, fields, batchSize, offset, orderby)**: Query a Profile data model object using filters and a search key.
- **queryProfileApi(dataModelName, id, childDataModelName, searchKey, filters, fields, batchSize, offset, orderby)**: Query a Profile data model object and a child object using filters and a search key.
getAllMetadata()
Get all metadata, including Calculated Insights, Engagement, Profile, and other objects, as well as their relationships to other objects.

API Version
52.0

Requires Chatter
No

Signature
public static ConnectApi.CdpQueryMetadataOutput getAllMetadata()

Return Value
Type: ConnectApi.CdpQueryMetadataOutput

getAllMetadata(entityType, entityCategory, entityName)
Get all metadata, filtering for entity type, category, and name.

API Version
54.0

Requires Chatter
No

Signature
public static ConnectApi.CdpQueryMetadataOutput getAllMetadata(String entityType, String entityCategory, String entityName)

Parameters
entityType
Type: String
Type of metadata entity requested. Valid values are `DataLakeObject`, `DataModelObject`, and `CalculatedInsight`. If unspecified, all types are returned.

`entityCategory`
Type: `String`
Category of the metadata entity. Valid values are `Profile`, `Engagement`, and `Related`. If unspecified, all category entities are returned.

`entityName`
Type: `String`
Metadata name of the entity, for example `UnifiedIndividual__dlm`. If unspecified, a complete list of entities is returned.

Return Value
Type: `ConnectApi.CdpQueryMetadataOutput`

`getInsightsMetadata()`
Get Insight metadata, including Calculated Insight objects, their dimensions and measures.

API Version
52.0

Requires Chatter
No

Signature
`public static ConnectApi.CdpQueryMetadataOutput getInsightsMetadata()`

Return Value
Type: `ConnectApi.CdpQueryMetadataOutput`

`getInsightsMetadata(ciName)`
Get metadata for a Calculated Insight object. Metadata includes dimensions and measures.

API Version
52.0

Requires Chatter
No

Signature
`public static ConnectApi.CdpQueryMetadataOutput getInsightsMetadata(String ciName)`
Parameters

ciName
Type: String
Name of the Calculated Insight object, for example, IndividualChildrenCount__cio.

Return Value
Type: ConnectApi.CdpQueryMetadataOutput

getProfileMetadata()
Get metadata for data model objects in the profile category, including Individual, Contact Point Email, Unified Individual, and Contact Point Address objects. Metadata includes the objects, their fields, and category.

API Version
52.0

Requires Chatter
No

Signature
public static ConnectApi.CdpQueryMetadataOutput getProfileMetadata()

Return Value
Type: ConnectApi.CdpQueryMetadataOutput

getProfileMetadata(dataModelName)
Get metadata for a data model object in the profile category, such as Individual, Contact Point Email, Unified Individual, and Contact Point Address. Metadata includes the list of fields, data types, and indexes available for lookup.

API Version
52.0

Requires Chatter
No

Signature
public static ConnectApi.CdpQueryMetadataOutput getProfileMetadata(String dataModelName)
Parameters

dataModelName
Type: String
Name of the data model object, for example, UnifiedIndividual__dlm.

Return Value
Type: `ConnectApi.CdpQueryMetadataOutput`

`nextBatchAnsiSqlV2(nextBatchId)`
Get the next batch of data across data model, lake, unified, and linked objects.

API Version
54.0

Requires Chatter
No

Signature
`public static ConnectApi.CdpQueryOutputV2 nextBatchAnsiSqlV2(String nextBatchId)`

Parameters

`nextBatchId`
Type: String
ID of the next batch. See the Usage section for more information.

Return Value
Type: `ConnectApi.CdpQueryOutputV2`

Usage
Initially, use the `queryAnsiSqlV2(input)` method to query up to 8 MB of data. Use the `nextBatchId` from the `ConnectApi.CdpQueryOutputV2` output class as the `nextBatchId` parameter in this method to get the next batch of data. You can continue using subsequent next batch IDs for up to an hour.

`queryAnsiSql(input)`
Synchronously query data across data model, lake, unified, and linked objects. This query returns up to 4,999 rows.

**Note:** A newer version of the Query API is available. We recommend using `queryAnsiSqlV2(input)` and `nextBatchAnsiSqlV2(nextBatchId)` to take advantage of subsequent requests and larger response sizes.
API Version
5.2.0

Requires Chatter
No

Signature
public static ConnectApi.CdpQueryOutput queryANSISql(ConnectApi.CdpQueryInput input)

Parameters
input
  Type: ConnectApi.CdpQueryInput
  A ConnectApi.CdpQueryInput body with the SQL query.

Return Value
Type: ConnectApi.CdpQueryOutput

queryANSISql(input, batchSize, offset, orderby)
Synchronously query data across data model, lake, unified, and linked objects. Specify batch size, offset, and order of the results. This query returns up to 4,999 rows.

Note: A newer version of the Query API is available. We recommend using queryAnsiSqlV2(input) and nextBatchAnsiSqlV2(nextBatchId) to take advantage of subsequent requests and larger response sizes.

API Version
5.3.0

Requires Chatter
No

Signature
public static ConnectApi.CdpQueryOutput queryANSISql(ConnectApi.CdpQueryInput input, Integer batchSize, Integer offset, String orderby)

Parameters
input
  Type: ConnectApi.CdpQueryInput
  A ConnectApi.CdpQueryInput body with the SQL query.

batchSize
  Type: Integer
Number of records to return. Values are from 1–4999. The default value is 4999.

_offset
Type: Integer
Number of rows to skip before returning results. The sum of offset and batchSize must be less than 2147483647. The default value is 0.

_orderby
Type: String
Comma-separated values to sort the results in ascending or descending order, for example, GenderId__c ASC, Occupation__c DESC.

Return Value
Type: ConnectApi.CdpQueryOutput

queryAnsiSqlV2(input)
Query up to 8 MB of data across data model, lake, unified, and linked objects.

API Version
54.0

Requires Chatter
No

Signature
public static ConnectApi.CdpQueryOutputV2 queryAnsiSqlV2(ConnectApi.CdpQueryInput input)

Parameters
input
Type: ConnectApi.CdpQueryInput
A ConnectApi.CdpQueryInput body with the SQL query.

Return Value
Type: ConnectApi.CdpQueryOutputV2

Usage
Use the nextBatchId in the ConnectApi.CdpQueryOutputV2 output class as the nextBatchId parameter in the nextBatchAnsiSqlV2(nextBatchId) method to continue getting batches of data for up to an hour.

queryCalculatedInsights(ciName, dimensions, measures, orderby, filters, batchSize, offset)
Query a Calculated Insight object.
API Version
$2.0

Requires Chatter
No

Signature

```java
public static ConnectApi.CdpQueryOutput queryCalculatedInsights(String ciName, String dimensions, String measures, String orderby, String filters, Integer batchSize, Integer offset)
```

Parameters

ciName
Type: String
Name of the Calculated Insight object, for example, IndividualChildrenCount__cio.

dimensions
Type: String
Comma-separated list of up to 10 dimensions, such as GenderId__c, to project. If unspecified, this parameter includes all of the available dimensions.

measures
Type: String
Comma-separated list of up to 5 measures, such as TotalSales__c, to project. If unspecified, this parameter includes all of the available measures.

orderby
Type: String
Sort order for the result set, such as GenderId__c ASC, Occupation__c DESC. If unspecified, items are returned in the order they are retrieved.

filters
Type: String
Filter the result set to a more narrow scope or specific type, such as [GenderId__c=Male, FirstName__c=Angel].

batchSize
Type: Integer
Number of items to return. Values are from 1–4,999. If unspecified, the default value is 4999.

offset
Type: Integer
Number of rows to skip before returning results. If unspecified, no rows are skipped.

Return Value
Type: ConnectApi.CdpQueryOutput
queryCalculatedInsights(ciName, dimensions, measures, orderby, filters, batchSize, offset, timeGranularity)

Query a Calculated Insight object within a specified time range.

API Version
54.0

Requires Chatter
No

Signature
public static ConnectApi.CdpQueryOutput queryCalculatedInsights(String ciName, String dimensions, String measures, String orderby, String filters, Integer batchSize, Integer offset, String timeGranularity)

Parameters

**ciName**
- Type: String
- Name of the Calculated Insight object, for example, IndividualChildrenCount__c.

**dimensions**
- Type: String
- Comma-separated list of up to 10 dimensions, such as GenderId__c, to project. If unspecified, this parameter includes all of the available dimensions.

**measures**
- Type: String
- Comma-separated list of up to 5 measures, such as TotalSales__c, to project. If unspecified, this parameter includes all of the available measures.

**orderby**
- Type: String
- Sort order for the result set, such as GenderId__c ASC, Occupation__c DESC. If unspecified, items are returned in the order they are retrieved.

**filters**
- Type: String
- Filter the result set to a more narrow scope or specific type, such as [GenderId__c=Male, FirstName__c=Angel].

**batchSize**
- Type: Integer
- Number of items to return. Values are from 1–4,999. If unspecified, the default value is 4999.

**offset**
- Type: Integer
- Number of rows to skip before returning results. If unspecified, no rows are skipped.
timeGranularity
Type: String
Time range for the measures. Values are:
- HOUR
- DAY
- MONTH
- QUARTER
- YEAR
If unspecified, no time range is applied.

Return Value
Type: ConnectApi.CdpQueryOutput

queryProfileApi(dataModelName, filters, fields, batchSize, offset, orderby)
Query a Profile data model object using filters.

API Version
52.0

Requires Chatter
No

Signature
public static ConnectApi.CdpQueryOutput queryProfileApi(String dataModelName, String filters, String fields, Integer batchSize, Integer offset, String orderby)

Parameters
dataModelName
Type: String
Name of the data model object, for example, UnifiedIndividual__d.

filters
Type: String
Comma-separated list of equality expressions within square brackets, for example, [FirstName__c=DON].

fields
Type: String
Comma-separated list of up to 50 field names that you want to include in the result, for example, Id__c, FirstName__c, GenderId__c, Occupation__c. If unspecified, Id__c is returned.
Number of items to return. Values are from 1–4,999. If unspecified, the default value is 100.

offset
  Type: Integer
  Number of rows to skip before returning results. If unspecified, no rows are skipped.

orderby
  Type: String
  Sort order for the result set, such as GenderId__c ASC, Occupation__c DESC. If unspecified, items are returned in the order they are retrieved.

Return Value
Type: ConnectApi.CdpQueryOutput

queryProfileApi(dataModelName, id, searchKey, filters, fields, batchSize, offset, orderby)
Query a Profile data model object using filters and a search key.

API Version
52.0

Requires Chatter
No

Signature
public static ConnectApi.CdpQueryOutput queryProfileApi(String dataModelName, String id, String searchKey, String filters, String fields, Integer batchSize, Integer offset, String orderby)

Parameters
dataModelName
  Type: String
  Name of the data model object, for example, UnifiedIndividual__dlm.

id
  Type: String
  Value of the primary or secondary key field, for example, John. If unspecified, defaults to the value of the primary key field.

searchKey
  Type: String
  If a field other than the primary key is used, name of the key field, for example, FirstName__c.

filters
  Type: String
  Comma-separated list of equality expressions within square brackets, for example, [FirstName__c=DON].
**fields**
  Type: **String**
  Comma-separated list of up to 50 field names that you want to include in the result, for example, `Id__c,FirstName__c, GenderId__c,Occupation__c`. If unspecified, `Id__c` is returned.

**batchSize**
  Type: **Integer**
  Number of items to return. Values are from 1–4,999. If unspecified, the default value is 100.

**offset**
  Type: **Integer**
  Number of rows to skip before returning results. If unspecified, no rows are skipped.

**orderby**
  Type: **String**
  Sort order for the result set, such as `GenderId__c ASC,Occupation__c DESC`. If unspecified, items are returned in the order they are retrieved.

**Return Value**
  Type: `ConnectApi.CdpQueryOutput`

```java
queryProfileApi(dataModelName, id, childDataModelName, searchKey, filters, fields, batchSize, offset, orderby)
```

Query a Profile data model object and a child object using filters and a search key.

**API Version**

52.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.CdpQueryOutput queryProfileApi(String dataModelName, String id, String childDataModelName, String searchKey, String filters, String fields, Integer batchSize, Integer offset, String orderby)
```

**Parameters**

**dataModelName**
  Type: **String**
  Name of the data model object, for example, `UnifiedIndividual__dlm`.

**id**
  Type: **String**
  Value of the primary or secondary key field, for example, `John`. If unspecified, defaults to the value of the primary key field.
childDataModelName
    Type: String
    Name of the child data model object, for example, UnifiedContactPointEmail__dlm.

searchKey
    Type: String
    If a field other than the primary key is used, name of the key field, for example, FirstName__c.

filters
    Type: String
    Comma-separated list of equality expressions within square brackets, for example, [FirstName__c=DON]. Filters are applied to the parent object only.

fields
    Type: String
    Comma-separated list of child object field names that you want to include in the result, for example, Id__c, EmailAddress__c. If unspecified, the first 10 alphabetically sorted fields are returned.

batchSize
    Type: Integer
    Number of items to return. Values are from 1–4,999. If unspecified, the default value is 100.

offset
    Type: Integer
    Number of rows to skip before returning results. If unspecified, no rows are skipped.

orderby
    Type: String
    Sort order for the result set, such as GenderId__c ASC, Occupation__c DESC. If unspecified, items are returned in the order they are retrieved.

Return Value
    Type: ConnectApi.CdpQueryOutput

queryProfileApi(dataModelName, id, ciName, searchKey, dimensions, measures, filters, fields, batchSize, offset, orderby)
Query a Profile data model object and a Calculated Insight object using filters and a search key.

API Version
    52.0

Requires Chatter
    No
**Signature**

```java
public static ConnectApi.CdpQueryOutput queryProfileApi(String dataModelName, String id, String ciName, String searchKey, String dimensions, String measures, String filters, String fields, Integer batchSize, Integer offset, String orderby)
```

**Parameters**

**dataModelName**

Type: `String`

Name of the data model object, for example, `UnifiedIndividual__dlm`.

**id**

Type: `String`

Value of the primary or secondary key field, for example, `John`. If unspecified, defaults to the value of the primary key field.

**ciName**

Type: `String`

Name of the Calculated Insight object, for example, `IndividualChildrenCount__cio`.

**searchKey**

Type: `String`

If a field other than the primary key is used, name of the key field, for example, `FirstName__c`.

**dimensions**

Type: `String`

Comma-separated list of up to 10 dimensions, such as `GenderId__c`, to project. If unspecified, this parameter includes all of the available dimensions.

**measures**

Type: `String`

Comma-separated list of up to 5 measures, such as `TotalSales__c`, to project. If unspecified, this parameter includes all of the available measures.

**filters**

Type: `String`

Comma-separated list of equality expressions within square brackets, for example, `[FirstName__c=DON]`.

**fields**

Type: `String`

Comma-separated list of up to 50 field names that you want to include in the result, for example, `Id__c,FirstName__c, GenderId__c, Occupation__c`. If unspecified, `Id__c` is returned.

**batchSize**

Type: `Integer`

Number of items to return. Values are from 1–4,999. If unspecified, the default value is 100.

**offset**

Type: `Integer`

Number of rows to skip before returning results. If unspecified, no rows are skipped.

**orderby**

Type: `String`
Sort order for the result set, such as GenderId__c ASC, Occupation__c DESC. If unspecified, items are returned in the order they are retrieved.

Return Value
Type: ConnectApi.CdpQueryOutput

queryProfileApi(dataModelName, id, ciName, searchKey, dimensions, measures, filters, fields, batchSize, offset, orderby, timeGranularity)
Query a Profile data model object and a Calculated Insight object using filters, a search key, and a time range.

API Version
54.0

Requires Chatter
No

Signature
public static ConnectApi.CdpQueryOutput queryProfileApi(String dataModelName, String id, String ciName, String searchKey, String dimensions, String measures, String filters, String fields, Integer batchSize, Integer offset, String orderby, String timeGranularity)

Parameters

dataModelName
Type: String
Name of the data model object, for example, UnifiedIndividual__dlm.

id
Type: String
Value of the primary or secondary key field, for example, John. If unspecified, defaults to the value of the primary key field.

ciName
Type: String
Name of the Calculated Insight object, for example, IndividualChildrenCount__cio.

searchKey
Type: String
If a field other than the primary key is used, name of the key field, for example, FirstName__c.

dimensions
Type: String
Comma-separated list of up to 10 dimensions, such as GenderId__c, to project. If unspecified, this parameter includes all of the available dimensions.

measures
Type: String
Comma-separated list of up to 5 measures, such as TotalSales__c, to project. If unspecified, this parameter includes all of the available measures.

**filters**

Type: String

Comma-separated list of equality expressions within square brackets, for example, [FirstName__c=DON].

**fields**

Type: String

Comma-separated list of up to 50 field names that you want to include in the result, for example, Id__c, FirstName__c, GenderId__c, Occupation__c. If unspecified, Id__c is returned.

**batchSize**

Type: Integer

Number of items to return. Values are from 1–4,999. If unspecified, the default value is 100.

**offset**

Type: Integer

Number of rows to skip before returning results. If unspecified, no rows are skipped.

**orderby**

Type: String

Sort order for the result set, such as GenderId__c ASC, Occupation__c DESC. If unspecified, items are returned in the order they are retrieved.

**timeGranularity**

Type: String

Time range for the measures. Values are:

- HOUR
- DAY
- MONTH
- QUARTER
- YEAR

If unspecified, no time range is applied.

**Return Value**

Type: ConnectApi.CdpQueryOutput

**universalIdLookupBySourceId(entityName, dataSourceId, dataSourceObjectId, sourceRecordId)**

Look up objects by source ID.

**API Version**

54.0
Requires Chatter
No

Signature
public static ConnectApi.CdpQueryDataOutput universalIdLookupBySourceId(String
entityName, String dataSourceId, String dataSourceObjectId, String sourceRecordId)

Parameters
entityName
  Type: String
  Entity name.
dataSourceId
  Type: String
  Data source ID.
dataSourceObjectId
  Type: String
  Data source object ID.
sourceRecordId
  Type: String
  Source record ID.

Return Value
Type: ConnectApi.CdpQueryDataOutput

CdpSegment Class
Create, delete, get, publish, and update Data Cloud segments. Get segment members.

Namespace
ConnectApi

CdpSegment Methods
The following are methods for CdpSegment. All methods are static.

IN THIS SECTION:
  createSegment(input)  
  Create a segment.
  createSegment(input, dataspace)  
  Create a segment in a dataspace.
deleteSegment(segmentApiName)
Delete a segment.

executePublishAdhoc(segmentId)
Publish a segment.

getSegment(segmentApiName)
Get a segment.

getSegmentMembers(limit, offset, orderBy, segmentApiName, filters, fields)
Get segment members.

getSegments()
Get segments.

getSegmentsPaginated(batchSize, offset, orderBy)
Get an ordered batch of segments.

getSegmentsPaginated(batchSize, offset, orderBy, dataspace)
Get paginated segments in a dataspace.

updateSegment(segmentApiName, input)
Update a segment.

**createSegment(input)**
Create a segment.

**API Version**
55.0

**Requires Chatter**
No

**Signature**

```
public static ConnectApi.CdpSegmentOutput createSegment(ConnectApi.CdpSegmentInput input)
```

**Parameters**

- **input**
  - Type: `ConnectApi.CdpSegmentInput`
  - A `ConnectApi.CdpSegmentInput` class.

**Return Value**

- Type: `ConnectApi.CdpSegmentOutput`
createSegment(input, dataspace)
Create a segment in a dataspace.

API Version
58.0

Requires Chatter
No

Signature
public static ConnectApi.CdpSegmentOutput createSegment(ConnectApi.CdpSegmentInput input, String dataspace)

Parameters
input
Type: ConnectApi.CdpSegmentInput
A ConnectApi.CdpSegmentInput class.
dataspace
Type: String
Name of the dataspace in which to perform the action. User must have permission to this dataspace. Optional when bulk-listing segments. If unspecified, default is the default dataspace.

Return Value
Type: ConnectApi.CdpSegmentOutput

deleteSegment(segmentApiName)
Delete a segment.

API Version
56.0

Requires Chatter
No

Signature
public static Void deleteSegment(String segmentApiName)
Parameters

segmentApiName
Type: String
API name of the segment.

Return Value
Type: Void

executePublishAdhoc(segmentId)
Publish a segment.

API Version
56.0

Requires Chatter
No

Signature
public static ConnectApi.CdpSegmentActionOutput executePublishAdhoc(String segmentId)

Parameters

segmentId
Type: String
ID of the segment to publish.

Return Value
Type: ConnectApi.CdpSegmentActionOutput

getSegment(segmentApiName)
Get a segment.

API Version
56.0

Requires Chatter
No

Signature
public static ConnectApi.CdpSegmentContainerOutput getSegment(String segmentApiName)
Parameters

segmentApiName
  Type: String
  API name of the segment.

Return Value

Type: ConnectApi.CdpSegmentContainerOutput

getSegmentMembers(limit, offset, orderBy, segmentApiName, filters, fields)
Get segment members.

API Version

58.0

Requires Chatter

No

Signature

public static ConnectApi.CdpSegmentMemberOutput getSegmentMembers(Integer limit, Integer offset, String orderBy, String segmentApiName, String filters, List<String> fields)

Parameters

limit
  Type: Integer
  Maximum number of rows to return from 1–1000. If unspecified, the default number of rows is 100.

offset
  Type: Integer
  Number of rows to skip before returning results. If unspecified, no rows are skipped.

orderBy
  Type: String
  Sort order for the result set, such as Name ASC or MarketSegmentType DESC. If unspecified, items are returned by ID in ascending order.

segmentApiName
  Type: String
  API name of the segment.

filters
  Type: String
  Filter the result set to a more narrow scope or specific type, such as [Delta_Type__C in('new') AND timestamp__C > '2023-04-02'].
**Fields**

Type: `List<String>`

Comma-separated list of up to 50 field names that you want to include in the result, for example, `Id__c`, `Delta_Type__C`.

If unspecified, `Id__c` is returned.

**Return Value**

Type: `ConnectApi.CdpSegmentMemberOutput`

getSegments()

Get segments.

**API Version**

55.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.CdpSegmentContainerOutput getSegments()
```

**Return Value**

Type: `ConnectApi.CdpSegmentContainerOutput`

getSegmentsPaginated(batchSize, offset, orderBy)

Get an ordered batch of segments.

**API Version**

56.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.CdpSegmentContainerOutput getSegmentsPaginated(Integer batchSize, Integer offset, String orderBy)
```

**Parameters**

`batchSize`

Type: `Integer`
Number of items to return. Values are from 1–200. If unspecified, the default value is 20.

offset
Type: Integer
Number of rows to skip before returning results. If unspecified, no rows are skipped.

orderBy
Type: String
Sort order for the result set, such as Name ASC or MarketSegmentType DESC. If unspecified, items are returned by ID in ascending order.

Return Value
Type: ConnectApi.CdpSegmentContainerOutput

getSegmentsPaginated(batchSize, offset, orderBy, dataspace)
Get paginated segments in a dataspace.

API Version
58.0

Requires Chatter
No

Signature
public static ConnectApi.CdpSegmentContainerOutput getSegmentsPaginated(Integer batchSize, Integer offset, String orderBy, String dataspace)

Parameters
batchSize
Type: Integer
Number of items to return. Values are from 1–200. If unspecified, the default value is 20.

offset
Type: Integer
Number of rows to skip before returning results. If unspecified, no rows are skipped.

orderBy
Type: String
Sort order for the result set, such as Name ASC or MarketSegmentType DESC. If unspecified, items are returned by ID in ascending order.

dataspace
Type: String
Name of the dataspace in which to perform the action. User must have permission to this dataspace. Optional when bulk-listing segments. If unspecified, default is the default dataspace.
Return Value
Type: ConnectApi.CdpSegmentContainerOutput

updateSegment(segmentApiName, input)
Update a segment.

API Version
56.0

Requires Chatter
No

Signature
public static ConnectApi.CdpSegmentOutput updateSegment(String segmentApiName, ConnectApi.CdpSegmentInput input)

Parameters
segmentApiName
Type: String
API name of the segment.

input
Type: ConnectApi.CdpSegmentInput
A ConnectApi.CdpSegmentInput class with the updates.

Return Value
Type: ConnectApi.CdpSegmentOutput

Chatter Class
Access information about followers and subscriptions for records.

Namespace
ConnectApi

Chatter Methods
The following are methods for Chatter. All methods are static.
IN THIS SECTION:

deleteSubscription(communityId, subscriptionId)
Delete a subscription. Use this method to stop following a record, a user, or a file.

getFollowers(communityId, recordId)
Get the first page of followers for a record.

getFollowers(communityId, recordId, pageParam, pageSize)
Get a page of followers for a record.

getSubscription(communityId, subscriptionId)
Get information about a subscription.

submitDigestJob(period)
Submit a daily or weekly Chatter email digest job.

deleteSubscription(communityId, subscriptionId)
Delete a subscription. Use this method to stop following a record, a user, or a file.

API Version
28.0

Requires Chatter
Yes

Signature
public static void deleteSubscription(String communityId, String subscriptionId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subscriptionId
Type: String
The ID for a subscription.

Return Value
Type: Void

Usage
“Following” a user, group, or record is the same as “subscribing” to a user, group, or record. A “follower” is the user who followed the user, group, or record. A “subscription” is an object describing the relationship between the follower and the user, group, or record they followed.

To leave a group, call deleteMember(communityId, membershipId).
Example

When you follow a user, the call to `ConnectApi.ChatterUsers.follow` returns a `ConnectApi.Subscription` object. To stop following the user, pass the `id` property of that object to this method.

```
ConnectApi.Chatter.deleteSubscription(null, '0E8RR0000004CnK0AU');
```

SEE ALSO:

- Follow a Record
- `follow(communityId, userId, subjectId)`

\textbf{getFollowers}(communityId, recordId)

Get the first page of followers for a record.

API Version

28.0

Requires Chatter

Yes

Signature

```
public static ConnectApi.FollowerPage getFollowers(String communityId, String recordId)
```

Parameters

- \textit{communityId}
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.
- \textit{recordId}
  - Type: `String`
  - ID for a record or the keyword `me`.

Return Value

Type: `ConnectApi.FollowerPage`

Usage

“Following” a user, group, or record is the same as “subscribing” to a user, group, or record. A “follower” is the user who followed the user, group, or record. A “subscription” is an object describing the relationship between the follower and the user, group, or record they followed.

SEE ALSO:

- Follow a Record
getFollowers(communityId, recordId, pageParam, pageSize)
Get a page of followers for a record.

API Version
28.0

Requires Chatter
Yes

Signature
public static ConnectApi.FollowerPage getFollowers(String communityId, String recordId,
Integer pageParam, Integer pageSize)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

recordId
Type: String
ID for a record or the keyword me.

pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

Return Value
Type: ConnectApi.FollowerPage

Usage
“Following” a user, group, or record is the same as “subscribing” to a user, group, or record. A “follower” is the user who followed the user, group, or record. A “subscription” is an object describing the relationship between the follower and the user, group, or record they followed.

SEE ALSO:
Follow a Record

getSubscription(communityId, subscriptionId)
Get information about a subscription.
API Version
28.0

Requires Chatter
Yes

Signature
public static ConnectApi.Subscription getSubscription(String communityId, String subscriptionId)

Parameters
communityId
- Type: String
- ID for an Experience Cloud site, internal, or null.

subscriptionId
- Type: String
- The ID for a subscription.

Return Value
Type: ConnectApi.Subscription

Usage
“Following” a user, group, or record is the same as “subscribing” to a user, group, or record. A “follower” is the user who followed the user, group, or record. A “subscription” is an object describing the relationship between the follower and the user, group, or record they followed.

SEE ALSO:
- Follow a Record

submitDigestJob(period)
Submit a daily or weekly Chatter email digest job.

API Version
37.0

Requires Chatter
Yes
Signature

```java
public static ConnectApi.DigestJobRepresentationsubmitDigestJob(ConnectApi.DigestPeriod period)
```

Parameters

```java
period
```
Type: `ConnectApi.DigestPeriod`

Time period that's included in a Chatter email digest. Values are:

- **DailyDigest**—The email includes up to the 50 latest posts from the previous day.
- **WeeklyDigest**—The email includes up to the 50 latest posts from the previous week.

Return Value

Type: `ConnectApi.DigestJob`

Usage

The times when Chatter sends email digests are not configurable in the UI. To control when email digests are sent and to use this method, contact Salesforce to enable API-only Chatter Digests.

⚠️ **Warning:** Enabling API-only Chatter Digests disables the scheduled digests for your org. You must call the API for your users to receive their digests.

We recommend scheduling digest jobs by implementing the Apex `Schedulable` interface with this method. To monitor your digest jobs from Setup, enter **Background Jobs** in the **Quick Find** box, then select **Background Jobs**.

Example

Schedule daily digests:

```java
global class ExampleDigestJob1 implements Schedulable {
    global void execute(SchedulableContext context) {
        ConnectApi.Chatter.submitDigestJob(ConnectApi.DigestPeriod.DailyDigest);
    }
}
```

Schedule weekly digests:

```java
global class ExampleDigestJob2 implements Schedulable {
    global void execute(SchedulableContext context) {
        ConnectApi.Chatter.submitDigestJob(ConnectApi.DigestPeriod.WeeklyDigest);
    }
}
```

SEE ALSO:
- Apex Scheduler

**ChatterFavorites Class**

Chatter favorites give you easy access to topics, list views, and feed searches.
Namespace

ConnectApi

Usage

Use Connect in Apex to get and delete topics, list views, and feed searches that have been added as favorites. Add topics and feed searches as favorites, and update the last view date of a feed search or list view feed to the current system time.

In this image of Salesforce, “Build Issues” is a topic, “All Accounts” is a list view, and “United” is a feed search.

ChatterFavorites Methods

The following are methods for ChatterFavorites. All methods are static.

IN THIS SECTION:

addFavorite(communityId, subjectId, searchText)
Add a feed search favorite for a user.

addRecordFavorite(communityId, subjectId, targetId)
Add a topic as a favorite.

deleteFavorite(communityId, subjectId, favoriteId)
Delete a favorite.

getFavorite(communityId, subjectId, favoriteId)
Get information about a favorite.

getFavorites(communityId, subjectId)
Get a list of favorites for a user.

getFeedElements(communityId, subjectId, favoriteId)
Get the first page of feed elements for a favorite.
getFeedElements(communityId, subjectId, favoritelid, pageParam, pageSize, sortParam)
Get a page of sorted feed elements for a favorite.

getFeedElements(communityId, subjectId, favoritelid, recentCommentCount, elementsPerBundle, pageParam, pageSize, sortParam)
Get a page of sorted feed elements for a favorite. Include no more than the specified number of comments per feed element.

updateFavorite(communityId, subjectId, favoritelid, updateLastViewDate)
Update the last view date of the saved search or list view feed to the current system time.

addFavorite(communityId, subjectId, searchText)
Add a feed search favorite for a user.

API Version
28.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedFavorite addFavorite(String communityId, String subjectId, String searchText)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subjectId
Type: String
ID of the context user or the alias me.

searchText
Type: String
Specify the text of the search to be saved as a favorite. This method can only create a feed search favorite, not a list view favorite or a topic.

Return Value
Type: ConnectApi.FeedFavorite

addRecordFavorite(communityId, subjectId, targetId)
Add a topic as a favorite.

API Version
28.0
Requires Chatter

Yes

Signature

public static ConnectApi.FeedFavorite addRecordFavorite(String communityId, String
subjectId, String targetId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subjectId
Type: String
ID of the context user or the alias me.

targetId
Type: String
The ID of the topic to add as a favorite.

Return Value

Type: ConnectApi.FeedFavorite

deleteFavorite(communityId, subjectId, favoriteId)

Delete a favorite.

API Version

28.0

Requires Chatter

Yes

Signature

public static Void deleteFavorite(String communityId, String subjectId, String
favoriteId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subjectId
Type: String
ID of the context user or the alias me.

*favoriteId*
Type: String
ID of a favorite.

**Return Value**
Type: Void

**getFavorite(communityId, subjectId, favoriteId)**
Get information about a favorite.

**API Version**
28.0

**Requires Chatter**
Yes

**Signature**
public static ConnectApi.FeedFavorite getFavorite(String communityId, String subjectId, String favoriteId)

**Parameters**

*communityId*
Type: String
ID for an Experience Cloud site, internal, or null.

*subjectId*
Type: String
ID of the context user or the alias me.

*favoriteId*
Type: String
ID of a favorite.

**Return Value**
Type: ConnectApi.FeedFavorite

**getFavorites(communityId, subjectId)**
Get a list of favorites for a user.
API Version
28.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedFavorites getFavorites(String communityId, String subjectId)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subjectId
Type: String
ID of the context user or the alias me.

Return Value
Type: ConnectApi.FeedFavorites

getFeedElements(communityId, subjectId, favoriteId)
Get the first page of feed elements for a favorite.

API Version
31.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedElementPage getFeedElements(String communityId, String subjectId, String favoriteId)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subjectId
Type: String
ID of the context user or the alias me.

favoriteId
Type: String
ID of a favorite.

Return Value
Type: ConnectApi.FeedElementPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetFeedElements(communityId, subjectId, favoriteId, result)

getFeedElements(communityId, subjectId, favoriteId, pageParam, pageSize, sortParam)
Get a page of sorted feed elements for a favorite.

API Version
31.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedElementPage getFeedElements(String communityId, String subjectId, String favoriteId, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subjectId
Type: String
ID of the context user or the alias me.

favoriteId
Type: String
ID of a favorite.

*pageParam*

Type: `String`

Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

*pageSize*

Type: `Integer`

Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

*sortParam*

Type: `ConnectApi.FeedSortOrder`

Values are:

- `CreatedDateAsc`—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- `CreatedDateDesc`—Sorts by most recent creation date.
- `LastModifiedDateDesc`—Sorts by most recent activity.
- `MostViewed`—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- `Relevance`—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.

If you pass in `null`, the default value `CreatedDateDesc` is used.

**Return Value**

Type: `ConnectApi.FeedElementPage`

**Usage**

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- `setTestGetFeedElements(communityId, subjectId, favoriteId, pageParam, pageSize, sortParam, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

```java
getFeedElements(communityId, subjectId, favoriteId, recentCommentCount, elementsPerBundle, pageParam, pageSize, sortParam)
```

Get a page of sorted feed elements for a favorite. Include no more than the specified number of comments per feed element.

**API Version**

31.0

**Requires Chatter**

Yes
Signature

```java
public static ConnectApi.FeedElementPage getFeedElements(String communityId, String subjectId, String favoriteId, Integer recentCommentCount, Integer elementsPerBundle, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam)
```

Parameters

`communityId`
- Type: String
- ID for an Experience Cloud site, internal, or null.

`subjectId`
- Type: String
- ID of the context user or the alias me.

`favoriteId`
- Type: String
- ID of a favorite.

`recentCommentCount`
- Type: Integer
- Maximum number of comments to return with each feed element. The default value is 3.

`elementsPerBundle`
- Type: Integer
- Maximum number of feed elements per bundle. The default and maximum value is 10.

`pageParam`
- Type: String
- Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

`pageSize`
- Type: Integer
- Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

`sortParam`
- Type: ConnectApi.FeedSortOrder
- Values are:
  - CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
  - CreatedDateDesc—Sorts by most recent creation date.
  - LastModifiedDateDesc—Sorts by most recent activity.
  - MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
  - Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
- If you pass in null, the default value CreatedDateDesc is used.
Return Value
Type: `ConnectApi.FeedElementPage`

Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestGetFeedElements(communityId, subjectId, favoriteId, recentCommentCount, elementsPerBundle, pageParam, pageSize, sortParam, result)`

`updateFavorite(communityId, subjectId, favoriteId, updateLastViewDate)`
Update the last view date of the saved search or list view feed to the current system time.

API Version
28.0

Requires Chatter
Yes

Signature
```java
public static ConnectApi.FeedFavorite updateFavorite(String communityId, String subjectId, String favoriteId, Boolean updateLastViewDate)
```

Parameters
- `communityId`
  Type: `String`
  ID for an Experience Cloud site, `internal`, or `null`.
- `subjectId`
  Type: `String`
  ID of the context user or the alias `me`.
- `favoriteId`
  Type: `String`
  ID of a favorite.
- `updateLastViewDate`
  Type: `Boolean`
  Specify whether to update the last view date of the specified favorite to the current system time (`true`) or not (`false`).
Return Value
Type: `ConnectApi.FeedFavorite`

ChatterFavorites Test Methods
The following are the test methods for `ChatterFavorites`. All methods are static.
For information about using these methods to test your `ConnectApi` code, see Testing ConnectApi Code.

IN THIS SECTION:

- `setTestGetFeedElements(communityId, subjectId, favoriteId, result)`
  Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElements` is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.
- `setTestGetFeedElements(communityId, subjectId, favoriteId, pageParam, pageSize, sortParam, result)`
  Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElements` is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.
- `setTestGetFeedElements(communityId, subjectId, favoriteId, recentCommentCount, elementsPerBundle, pageParam, pageSize, sortParam, result)`
  Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElements` is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.
- `setTestGetFeedElements(communityId, subjectId, favoriteId, result)`
  Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElements` is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

```java
public static void setTestGetFeedElements(String communityId, String subjectId, String favoriteId, ConnectApi.FeedElementPage result)
```

Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElements` is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

API Version
31.0

Signature

```java
public static void setTestGetFeedElements(String communityId, String subjectId, String favoriteId, ConnectApi.FeedElementPage result)
```

Parameters

- `communityId`
  Type: `String`
  ID for an Experience Cloud site, `internal`, or `null`.
- `subjectId`
  Type: `String`
  ID of the context user or the alias `me`.
- `favoriteId`
  Type: `String`
  ID of a favorite.
result
Type: ConnectApi.FeedElementPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
getFeedElements(communityId, subjectId, favoriteId)

setTestGetFeedElements(communityId, subjectId, favoriteId, pageParam, pageSize, sortParam, result)
Register a ConnectApi.FeedElementPage object to be returned when getFeedElements is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

API Version
31.0

Signature
public static Void setTestGetFeedElements(String communityId, String subjectId, String favoriteId, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedElementPage result)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subjectId
Type: String
ID of the context user or the alias me.

favoriteId
Type: String
ID of a favorite.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**

Type: `ConnectApi.FeedSortOrder`

Values are:

- `CreatedDateAsc`—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- `CreatedDateDesc`—Sorts by most recent creation date.
- `LastModifiedDateDesc`—Sorts by most recent activity.
- `MostViewed`—Sorts by most viewed content. This sort order is available only for Home feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- `Relevance`—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

If you pass in `null`, the default value `CreatedDateDesc` is used.

**result**

Type: `ConnectApi.FeedElementPage`

Object containing test data.

**Return Value**

Type: Void

SEE ALSO:

- `getFeedElements(communityId, subjectId, favoriteId, pageParam, pageSize, sortParam)`
  - *Apex Developer Guide: Testing ConnectApi Code*

**setTestGetFeedElements(communityId, subjectId, favoriteId, recentCommentCount, elementsPerBundle, pageParam, pageSize, sortParam, result)**

Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElements` is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

**API Version**

31.0

**Signature**

```java
public static Void setTestGetFeedElements(String communityId, String subjectId, String favoriteId, Integer recentCommentCount, Integer elementsPerBundle, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedElementPage result)
```

**Parameters**

- **communityId**
  
  Type: `String`

  ID for an Experience Cloud site, internal, or `null`.  

subjectId
Type: String
ID of the context user or the alias me.

favoriteId
Type: String
ID of a favorite.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

elementsPerBundle
Type: Integer
Maximum number of feed elements per bundle. The default and maximum value is 10.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
• CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
• CreatedDateDesc—Sorts by most recent creation date.
• LastModifiedDateDesc—Sorts by most recent activity.
• MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
• Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
If you pass in null, the default value CreatedDateDesc is used.

result
Type: ConnectApi.FeedElementPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
getFeedElements(communityId, subjectId, favoriteId, recentCommentCount, elementsPerBundle, pageParam, pageSize, sortParam)

Retired ChatterFavorites Methods

The following methods for ChatterFavorites are retired.

IN THIS SECTION:

- `getFeedItems(communityId, subjectId, favoriteId)`
  - Get the first page of feed items for a favorite.

- `getFeedItems(communityId, subjectId, favoriteId, pageParam, pageSize, sortParam)`
  - Get a page of sorted feed items for a favorite.

- `getFeedItems(communityId, subjectId, favoriteId, recentCommentCount, pageParam, pageSize, sortParam)`
  - Get a page of sorted feed items for a favorite. Include no more than the specified number of comments per feed item.

- `setTestGetFeedItems(communityId, subjectId, favoriteId, result)`
  - Register a `ConnectApi.FeedItemPage` object to be returned when `getFeedItems` is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

- `setTestGetFeedItems(communityId, subjectId, favoriteId, pageParam, pageSize, sortParam, result)`
  - Register a `ConnectApi.FeedItemPage` object to be returned when `getFeedItems` is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

- `setTestGetFeedItems(communityId, subjectId, favoriteId, recentCommentCount, pageParam, pageSize, sortParam, result)`
  - Register a `ConnectApi.FeedItemPage` object to be returned when `getFeedItems` is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

### `getFeedItems(communityId, subjectId, favoriteId)`

Get the first page of feed items for a favorite.

**API Version**

28.0–31.0

ℹ️ **Important:** In version 32.0 and later, use `getFeedElements(communityId, subjectId, favoriteId)`.

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.FeedItemPage getFeedItems(String communityId, String subjectId, String favoriteId)
```

**Parameters**

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, `internal`, or `null`.

---

450
**subjectId**
Type: **String**
ID of the context user or the alias `me`.

**favoriteId**
Type: **String**
ID of a favorite.

**Return Value**
Type: `ConnectApi.FeedItemPage`

**Usage**
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

**SEE ALSO:**
- `setTestGetFeedItems(communityId, subjectId, favoriteId, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

**getFeedItems**(communityId, subjectId, favoriteId, pageParam, pageSize, sortParam)
Get a page of sorted feed items for a favorite.

**API Version**
28.0–31.0

**Important:** In version 32.0 and later, use `getFeedElements(communityId, subjectId, favoriteId, pageParam, pageSize, sortParam)`.

**Requires Chatter**
Yes

**Signature**
```java
public static ConnectApi.FeedItemPage getFeedItems(String communityId, String subjectId, String favoriteId, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam)
```

**Parameters**

**communityId**
Type: **String**
ID for an Experience Cloud site, `internal`, or `null`.
**subjectId**
Type: String
ID of the context user or the alias me.

**favoriteId**
Type: String
ID of a favorite.

**pageParam**
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

**pageSize**
Type: Integer
Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

**sortParam**
Type: ConnectApi.FeedSortOrder
Values are:
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null, the default value CreatedDateDesc is used.

**Return Value**
Type: ConnectApi.FeedItemPage

**Usage**
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- setTestGetFeedItems(communityId, subjectId, favoriteId, pageParam, pageSize, sortParam, result)
  *Apex Developer Guide: Testing ConnectApi Code*

**getFeedItems(communityId, subjectId, favoriteId, recentCommentCount, pageParam, pageSize, sortParam)**
Get a page of sorted feed items for a favorite. Include no more than the specified number of comments per feed item.
API Version
29.0–31.0

Important: In version 32.0 and later, use `getFeedElements(communityId, subjectId, favoriteId, recentCommentCount, elementsPerBundle, pageParam, pageSize, sortParam).

Requires Chatter
Yes

Signature
```
public static ConnectApi.FeedItemPage getFeedItems(String communityId, String subjectId, String favoriteId, Integer recentCommentCount, String pageParam, Integer pageSize, FeedSortOrder sortParam)
```

Parameters

`communityId`
Type: String
ID for an Experience Cloud site, internal, or `null`. 

`subjectId`
Type: String
ID of the context user or the alias `me`. 

`favoriteId`
Type: String
ID of a favorite. 

`recentCommentCount`
Type: Integer
Maximum number of comments to return with each feed item. The default value is 3. 

`pageParam`
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned. 

`pageSize`
Type: Integer
Number of feed items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25. 

`sortParam`
Type: `ConnectApi.FeedSortOrder`
Values are:
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds. 
- CreatedDateDesc—Sorts by most recent creation date. 
- LastModifiedDateDesc—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.

- **Relevance**—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds. Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in `null`, the default value `CreatedDateDesc` is used.

**Return Value**

Type: `ConnectApi.FeedItemPage`

**Usage**

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- `setTestGetFeedItems(communityId, subjectId, favoriteId, recentCommentCount, pageParam, pageSize, sortParam, result)`
  *Apex Developer Guide: Testing ConnectApi Code*

**setTestGetFeedItems(communityId, subjectId, favoriteId, result)**

Register a `ConnectApi.FeedItemPage` object to be returned when `getFeedItems` is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

**API Version**

28.0–31.0

**Signature**

```
public static Void setTestGetFeedItems(String communityId, String subjectId, String favoriteId, ConnectApi.FeedItemPage result)
```

**Parameters**

`communityId`

Type: `String`

ID for an Experience Cloud site, internal, or `null`.

`subjectId`

Type: `String`

ID of the context user or the alias `me`.

`favoriteId`

Type: `String`

ID of a favorite.

`result`

Type: `ConnectApi.FeedItemPage`
Object containing test data.

**Return Value**
Type: Void

SEE ALSO:
- `getFeedItems(communityId, subjectId, favoriteId)`
- *Apex Developer Guide: Testing ConnectApi Code*

### setTestGetFeedItems(communityId, subjectId, favoriteId, pageParam, pageSize, sortParam, result)

Register a `ConnectApi.FeedItemPage` object to be returned when `getFeedItems` is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

**API Version**
28.0–31.0

**Signature**
```
public static Void setTestGetFeedItems(String communityId, String subjectId, String favoriteId, String pageParam, Integer pageSize, FeedSortOrder sortParam, ConnectApi.FeedItemPage result)
```

**Parameters**
- **communityId**
  Type: String
  ID for an Experience Cloud site, internal, or `null`.
- **subjectId**
  Type: String
  ID of the context user or the alias `me`.
- **favoriteId**
  Type: String
  ID of a favorite.
- **pageParam**
  Type: String
  Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.
- **pageSize**
  Type: Integer
  Number of feed items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.
sortParam
Type: ConnectApi.FeedSortOrder
Values are:
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null, the default value CreatedDateDesc is used.

result
Type: ConnectApi_FEEDITEMPAGE
Object containing test data.

Return Value
Type: Void

SEE ALSO:
- getFeedItems(communityId, subjectId, favoriteId, pageParam, pageSize, sortParam)

setTestGetFeedItems(communityId, subjectId, favoriteId, recentCommentCount, pageParam, pageSize, sortParam, result)
Register a ConnectApi_FEEDITEMPAGE object to be returned when getFeedItems is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

API Version
29.0–31.0

Signature
public static Void setTestGetFeedItems(String communityId, String subjectId, String favoriteId, Integer recentCommentCount, String pageParam, Integer pageSize, FeedSortOrder sortParam, ConnectApi.FeedItemPage result)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.
subjectId
Type: String
ID of the context user or the alias me.

favoriteId
Type: String
ID of a favorite.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed item. The default value is 3.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null, the default value CreatedDateDesc is used.

result
Type: ConnectApi.FeedItemPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
- getFeedItems(communityId, subjectId, favoriteId, recentCommentCount, pageParam, pageSize, sortParam)
ChatterFeeds Class

Get, post, and delete feed elements, likes, comments, and bookmarks. You can also search feed elements, share feed elements, and vote on polls.

Namespace

ConnectApi

Usage

The Chatter feed is a container of feed elements. The abstract class ConnectApi.FeedElement is a parent class to the ConnectApi.FeedItem class, representing feed posts, and the ConnectApi.GenericFeedElement class, representing bundles and recommendations in the feed. For detailed information, see Working with Feeds and Feed Elements.

⚠️ Important: Feed item methods aren’t available in version 32.0. In version 32.0 and later, use feed element methods.

Message segments in a feed item are typed as ConnectApi.MessageSegment. Feed item capabilities are typed as ConnectApi.FeedItemCapability. Record fields are typed as ConnectApi.AbstractRecordField. These classes are all abstract and have several concrete subclasses. At runtime you can use instanceof to check the concrete types of these objects and then safely proceed with the corresponding downcast. When you downcast, you must have a default case that handles unknown subclasses.

⚠️ Important: The composition of a feed can change between releases. Write your code to handle instances of unknown subclasses.

ChatterFeeds Methods

The following are methods for ChatterFeeds. All methods are static.

IN THIS SECTION:

createStream(communityId, streamInput)
Create a Chatter feed stream.

deleteComment(communityId, commentId)
Delete a comment.

deleteFeedElement(communityId, feedElementId)
Delete a feed element.

deleteLike(communityId, likeld)
Delete a like on a comment or post.

deleteStream(communityId, streamId)
Delete a Chatter feed stream.

getComment(communityId, commentId)
Get a comment.

getCommentBatch(communityId, commentIds)
Get a list of comments.

getCommentInContext(communityId, commentId, pageSize)
Get a threaded comment in the context of its parent comments and post.
getCommentsForFeedElement(communityId, feedElementId)
Get comments for a feed element.

getCommentsForFeedElement(communityId, feedElementId, threadedCommentsCollapsed)
Get comments in a threaded style for a feed element.

getCommentsForFeedElement(communityId, feedElementId, pageParam, pageSize)
Get a page of comments for a feed element.

getCommentsForFeedElement(communityId, feedElementId, pageParam, pageSize, threadedCommentsCollapsed)
Get a page of comments in a threaded style for a feed element.

getCommentsForFeedElement(communityId, feedElementId, pageParam, pageSize, threadedCommentsCollapsed, sortParam)
Get a page of sorted comments in a threaded style for a feed element.

getCommentsForFeedElement(communityId, feedElementId, sortParam)
Get sorted comments for a feed element.

getCommentsForFeedElement(communityId, feedElementId, sortParam, threadedCommentsCollapsed)
Get sorted comments in a threaded style for a feed element.

getExtensions(communityId, pageParam, pageSize)
Get extensions.

getFeed(communityId, feedType)
Get a feed.

getFeed(communityId, feedType, sortParam)
Get a sorted feed.

getFeed(communityId, feedType, subjectId)
Get a feed for a record or user.

getFeed(communityId, feedType, subjectId, sortParam)
Get a sorted feed for a record or user.

getFeedDirectory(String)
Get a list of all feeds available to the context user.

getFeedElement(communityId, feedElementId)
Get a feed element.

getFeedElement(communityId, feedElementId, commentSort)
Get a feed element with sorted comments.

getFeedElement(communityId, feedElementId, threadedCommentsCollapsed)
Get a feed element and its comments in a threaded style.

getFeedElement(communityId, feedElementId, threadedCommentsCollapsed, commentSort)
Get a feed element and its sorted comments in a threaded style.

getFeedElement(communityId, feedElementId, recentCommentCount, elementsPerBundle)
Get a feed element with the specified number of elements per bundle including no more than the specified number of comments per feed element.
getFeedElement(communityId, feedElementId, recentCommentCount, elementsPerBundle, threadedCommentsCollapsed)
Get a feed element with its comments in a threaded style with the specified number of elements per bundle and comments per feed element.

getFeedElement(communityId, feedElementId, recentCommentCount, elementsPerBundle, threadedCommentsCollapsed, commentSort)
Get a feed element with its sorted comments in a threaded style with the specified number of elements per bundle and comments per feed element.

getFeedElement(communityId, feedElementId, recentCommentCount, elementsPerBundle, commentSort)
Get a feed element with the specified number of elements per bundle including no more than the specified number of sorted comments per feed element.

getFeedElementBatch(communityId, feedElementIds)
Get a list of feed elements.

getFeedElementPoll(communityId, feedElementId)
Get the poll associated with a feed element.

getFeedElementsFromBundle(communityId, feedElementId)
Get feed elements from a bundle.

getFeedElementsFromBundle(communityId, feedElementId, pageParam, pageSize, elementsPerBundle, recentCommentCount)
Get a page of feed elements from a bundle. Specify the number of elements per bundle and include no more than the specified number of comments per feed element.

getFeedElementsFromFeed(communityId, feedType)

getFeedElementsFromFeed(communityId, feedType, pageParam, pageSize, sortParam)

getFeedElementsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam)
Get a page of sorted feed elements from the Company, DirectMessageModeration, DirectMessages, Home, Moderation, and PendingReview feeds. Each feed element contains no more than the specified number of comments.

getFeedElementsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, filter)
Get a page of sorted and filtered feed elements from the Home feed. Each feed element contains no more than the specified number of comments.

getFeedElementsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, filter, threadedCommentsCollapsed)
Get a page of filtered and sorted feed elements with comments in a threaded style from the Home feed. Each feed element contains no more than the specified number of comments.

getFeedElementsFromFeed(communityId, feedType, subjectId)
Get feed elements from any feed other than Company, DirectMessageModeration, DirectMessages, Filter, Home, Landing, Moderation, and PendingReview for a user or record.

getFeedElementsFromFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam)
Get a page of sorted feed elements from any feed other than Company, DirectMessageModeration, DirectMessages, Filter, Home, Landing, Moderation, and PendingReview.
getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam)
Get a page of sorted feed elements from any feed other than Company, DirectMessageModeration, DirectMessages, Filter, Home, Landing, Moderation, and PendingReview. Each feed element includes no more than the specified number of comments.

getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, showInternalOnly)
Get a page of sorted feed elements from a record feed. Each feed element includes no more than the specified number of comments. Specify whether to return feed elements posted by internal (non-Experience Cloud site) users only.

getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, filter)
Get a page of sorted and filtered feed elements from the UserProfile feed.

getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, filter, threadedCommentsCollapsed)
Get a page of feed elements with comments in a threaded style from the UserProfile feed.

getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, customFilter)
Get a page of sorted and filtered feed elements from the case feed.

getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly)
Get a page of sorted feed elements from a record feed. Specify the number of elements per bundle and include no more than the specified number of comments per feed element. Specify whether to return feed elements posted by internal (non-Experience Cloud site) users only.

getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly, filter)
Get a page of sorted and filtered feed elements from a record feed. Specify the number of elements per bundle and include no more than the specified number of comments per feed element. Specify whether to return feed elements posted by internal (non-Experience Cloud site) users only.

getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly, filter, threadedCommentsCollapsed)
Get a page of sorted and filtered feed elements with comments in a threaded style from a record feed. Specify the number of elements per bundle and include no more than the specified number of comments per feed element. Specify whether to return feed elements posted by internal (non-Experience Cloud site) users only.

getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly, customFilter)
Get a page of sorted and filtered feed elements from a case feed.

getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly, customFilter, threadedCommentsCollapsed)
Get a page of filtered and sorted feed elements with comments in a threaded style from a case feed.

generateFeedElementsFromFilterFeed(communityId, subjectId, keyPrefix)
Get feed elements from a feed filtered by a key prefix for a user.

generateFeedElementsFromFilterFeed(communityId, subjectId, keyPrefix, pageParam, pageSize, sortParam)
Get a page of sorted feed elements from a feed filtered by a key prefix for a user.
getFeedElementsFromFilterFeed(communityId, subjectId, keyPrefix, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam)

Get a page of sorted feed elements from a feed filtered by a key prefix for a user. Each feed element contains no more than the specified number of comments.

getFeedElementsFromFilterFeedUpdatedSince(communityId, subjectId, keyPrefix, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince)

Get a page of feed elements from a feed filtered by a key prefix for a user. Include only feed elements that have been updated since the time specified in the updatedSince parameter. Each feed element contains no more than the specified number of comments.

getFeedElementsUpdatedSince(communityId, feedType, recentCommentCount, density, pageParam, pageSize, updatedSince)

Get a page of feed elements from the Company, DirectMessageModeration, Home, and Moderation feeds. Include only feed elements that have been updated since the time specified in the updatedSince parameter. Each feed element contains no more than the specified number of comments.

getFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince)

Get a page of feed elements from the Files, Groups, News, People, and Record feeds. Include only feed elements that have been updated since the time specified in the updatedSince parameter. Each feed element contains no more than the specified number of comments.

getFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince, filter)

Get a page of filtered feed elements from the Home feed. Include only feed elements that have been updated since the time specified in the updatedSince parameter. Each feed element contains no more than the specified number of comments.

getFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince)

Get a page of filtered feed elements from a UserProfile feed. Include only feed elements that have been updated since the time specified in the updatedSince parameter.

getFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, showInternalOnly)

Get a page of filtered feed elements from a case feed. Include only feed elements that have been updated since the time specified in the updatedSince parameter. Specify whether to return feed elements posted by internal (non-Experience Cloud site) users only.

getFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, customFilter)

Get a page of filtered feed elements from a case feed. Include only feed elements that have been updated since the time specified in the updatedSince parameter. Specify the maximum number of feed elements in a bundle.

getFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, showInternalOnly)

Get a page of filtered feed elements from a record feed. Include only feed elements that have been updated since the time specified in the updatedSince parameter. Specify whether to return feed elements posted by internal (non-Experience Cloud site) users only.

getFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, filter)

Get a page of filtered feed elements from a record feed. Include only feed elements that have been updated since the time specified in the updatedSince parameter. Specify the maximum number of feed elements in a bundle and whether to return feed elements posted by internal (non-Experience Cloud site) users only.
getFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, showInternalOnly, customFilter)
Get a page of filtered feed elements from a case feed. Include only feed elements that have been updated since the time specified in the updatedSince parameter.

getFeedWithFeedElements(communityId, feedType, pageSize)
Get information about a feed and a page of feed elements from the feed.

getFeedWithFeedElements(communityId, feedType, pageSize, recentCommentCount)
Get a page of information about the feed and the feed elements with the specified number of comments per feed element from the feed.

getFilterFeed(communityId, subjectId, keyPrefix)
Get a feed filtered by a key prefix for a user.

getFilterFeed(communityId, subjectId, keyPrefix, sortParam)
Get a sorted feed filtered by a key prefix for a user.

getFilterFeedDirectory(communityId, subjectId)
Get a feed directory of filter feeds available to the context user.

getLike(communityId, likeId)
Get a like on a post or comment.

getLikesForComment(communityId, commentId)
Get likes for a comment.

getLikesForComment(communityId, commentId, pageParam, pageSize)
Get a page of likes for a comment.

getLikesForFeedElement(communityId, feedElementId)
Get likes for a feed element.

getLikesForFeedElement(communityId, feedElementId, pageParam, pageSize)
Get a page of likes for a feed element.

getLinkMetadata(communityId, urls)
Get link metadata for URLs.

getPinnedFeedElementsFromFeed(communityId, feedType, subjectId)
Get pinned feed elements from a group or topic feed.

getReadByForFeedElement(communityId, feedElementId)
Get information about who read a feed element and when.

getReadByForFeedElement(communityId, feedElementId, pageParam, pageSize)
Get a page of information about who read a feed element and when.

getRelatedPosts(communityId, feedElementId, filter, maxResults)
Get posts related to the context feed element.

getStream(communityId, streamId)
Get information about a Chatter feed stream.

getStream(communityId, streamId, globalScope)
Get information about a Chatter feed stream, regardless of Experience Cloud site.

getStreams(communityId)
Get the Chatter feed streams for the context user.
getStreams(communityId, sortParam)
Get and sort the Chatter feed streams for the context user.

getStreams(communityId, pageParam, pageSize)
Get a page of Chatter feed streams for the context user.

getStreams(communityId, pageParam, pageSize, sortParam)
Get a sorted page of Chatter feed streams for the context user.

getStreams(communityId, pageParam, pageSize, sortParam, globalScope)
Get a sorted page of Chatter feed streams from all Enterprise Cloud sites for the context user.

getSupportedEmojis()
Get supported emojis for the org.

getThreadsForFeedComment(communityId, commentId)
Get threaded comments for a comment.

getThreadsForFeedComment(communityId, commentId, pageParam, pageSize)
Get a page of threaded comments for a comment.

getThreadsForFeedComment(communityId, commentId, threadedCommentsCollapsed)
Access the comments capability for a comment.

getTopUnansweredQuestions(communityId) (Pilot)
Get top unanswered questions for the context user in a Experience Cloud site.

getTopUnansweredQuestions(communityId, filter) (Pilot)
Get filtered top unanswered questions for the context user in an Experience Cloud site.

getTopUnansweredQuestions(communityId, pageSize) (Pilot)
Get a page of top unanswered questions for the context user in an Experience Cloud site.

getTopUnansweredQuestions(communityId, filter, pageSize) (Pilot)
Get a page of filtered top unanswered questions for the context user in an Experience Cloud site.

getVotesForComment(communityId, commentId, vote)
Get the first page of users who upvoted or downvoted a comment.

getVotesForComment(communityId, commentId, vote, pageParam, pageSize)
Get a page of users who upvoted or downvoted a comment.

getVotesForFeedElement(communityId, feedElementId, vote)
Get the first page of users who upvoted or downvoted a feed element.

getVotesForFeedElement(communityId, feedElementId, vote, pageParam, pageSize)
Get a page of users who upvoted or downvoted a feed element.

isCommentEditableByMe(communityId, commentId)
Discover whether the context user can edit a comment.

isFeedElementEditableByMe(communityId, feedElementId)
Discover whether the context user can edit a feed element.

isModified(communityId, feedType, subjectId, since)
Discover whether a news feed has been updated or changed. Use this method to poll a news feed for updates.

likeComment(communityId, commentId)
Like a comment for the context user.
likeFeedElement(communityId, feedElementId)
Like a feed element.

postCommentToFeedElement(communityId, feedElementId, text)
Post a plain-text comment to a feed element.

postCommentToFeedElement(communityId, feedElementId, comment, feedElementFileUpload)
Post a rich-text comment to a feed element. Use this method to include mentions and to attach a file.

postFeedElement(communityId, subjectId, feedElementType, text)
Post a plain-text feed element.

postFeedElement(communityId, feedElement)
Post a rich-text feed element. Include mentions and hashtag topics, attach already uploaded files to a feed element, and associate action link groups with a feed element. You can also use this method to share a feed element and add a comment.

postFeedElementBatch(communityId, feedElements)
Post a list of feed elements.

publishDraftFeedElement(communityId, feedElementId, feedElement)
Publish a draft feed element.

searchFeedElements(communityId, q)
Get the first page of feed elements that match the search criteria.

searchFeedElements(communityId, q, sortParam)
Get the first page of sorted feed elements that match the search criteria.

searchFeedElements(communityId, q, threadedCommentsCollapsed)
Get the feed elements and comments that match the search criteria.

searchFeedElements(communityId, q, pageParam, pageSize)
Get a page of feed elements that match the search criteria.

searchFeedElements(communityId, q, pageParam, pageSize, sortParam)
Get a page of sorted feed elements that match the search criteria.

searchFeedElements(communityId, q, pageParam, pageSize, threadedCommentsCollapsed)
Get a page of feed elements with comments in a threaded style that match the search criteria.

searchFeedElements(communityId, q, recentCommentCount, pageParam, pageSize, sortParam)
Get a page of sorted feed elements that match the search criteria. Each feed element includes no more than the specified number of comments.

searchFeedElementsInFeed(communityId, feedType, q)
Get the feed elements from the Company, DirectMessageModeration, Home, Moderation, and PendingReview feeds that match the search criteria.

searchFeedElementsInFeed(communityId, feedType, pageParam, pageSize, sortParam, q)
Get a page of sorted feed elements from the Company, DirectMessageModeration, Home, Moderation, and PendingReview feeds that match the search criteria.

searchFeedElementsInFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, q)
Get a page of sorted feed elements from the Company, DirectMessageModeration, Home, Moderation, and PendingReview feeds that match the search criteria. Each feed element includes no more than the specified number of comments.
searchFeedElementsInFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, q, filter)
Get a page of sorted and filtered feed elements from the Home feed that match the search criteria. Each feed element includes no more than the specified number of comments.

searchFeedElementsInFeed(communityId, feedType, subjectId, q)
Search up to 5,000 of the most recent feed elements in a feed for a subject ID that match the search string. Feed elements are returned in order of most recent activity.

searchFeedElementsInFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam, q)
Get a page of sorted feed elements from a feed for a record or user that match the search criteria.

searchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q)
Get a page of sorted feed elements from a feed that match the search criteria. Each feed element includes no more than the specified number of comments.

searchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, filter)
Get a page of sorted and filtered feed elements from a UserProfile feed that match the search criteria.

searchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, customFilter)
Get a page of sorted and filtered feed elements from a case feed that match the search criteria.

searchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, showInternalOnly)
Get a page of sorted feed elements from a feed for a record or user that match the search criteria. Each feed element includes no more than the specified number of comments. Specify whether to return feed elements posted by internal (non-Experience Cloud site) users only.

searchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, showInternalOnly, filter)
Get a page of sorted and filtered feed elements from a feed for a record or user that match the search criteria. Each feed element includes no more than the specified number of comments. Specify whether to return feed elements posted by internal (non-Experience Cloud site) users only.

searchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, showInternalOnly, customFilter)
Get a page of sorted and filtered feed elements from a case feed that match the search criteria.

searchFeedElementsInFilterFeed(communityId, subjectId, keyPrefix, q)
Get the feed elements from a feed filtered by a key prefix that match the search criteria.

searchFeedElementsInFilterFeed(communityId, subjectId, keyPrefix, pageParam, pageSize, sortParam, q)
Get a page of sorted feed elements from a feed filtered by a key prefix that match the search criteria.

searchFeedElementsInFilterFeed(communityId, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, sortParam, q)
Get a page of sorted feed elements from a feed filtered by a key prefix that match the search criteria. Each feed element includes no more than the specified number of comments.

searchStreams(communityId, q)
Search the Chatter feed streams for the context user.

searchStreams(communityId, q, sortParam)
Search and sort the Chatter feed streams for the context user.
searchStreams(communityId, q, pageParam, pageSize)
Search the Chatter feed streams for the context user and return a page of results.

searchStreams(communityId, q, pageParam, pageSize, sortParam)
Search the Chatter feed streams for the context user and return a sorted page of results.

searchStreams(communityId, q, pageParam, pageSize, sortParam, globalScope)
Search the Chatter feed streams from all Experience Cloud sites for the context user and return a sorted page of results.

setCommentIsVerified(communityId, commentId, isVerified)
Mark a comment as verified or unverified.

setCommentIsVerifiedByAnonymized(communityId, commentId, isVerified, isVerifiedByAnonymized)
Mark a comment as verified by an anonymous user.

setCommentVote(communityId, commentId, upDownVote)
Upvote or downvote a comment.

setFeedCommentStatus(communityId, commentId, status)
Set the status of a comment.

setFeedElementIsClosed(communityId, feedElementId, isClosed)
Set a feed element to closed.

setFeedElementVote(communityId, feedElementId, upDownVote)
Upvote or downvote a feed element.

setFeedEntityStatus(communityId, feedElementId, status)
Set the status of a feed post.

setIsMutedByMe(communityId, feedElementId, isMutedByMe)
Mute or unmute a feed element.

setIsReadByMe(communityId, feedElementId, isReadByMe)
Mark a feed element as read for the context user.

setIsReadByMe(communityId, feedElementId, readBy)
Mark a feed element as read for the context user using an input class.

updateComment(communityId, commentId, comment)
Edit a comment.

updateDirectMessage(communityId, feedElementId, directMessage)
Update the members of a direct message.

updateFeedElement(communityId, feedElementId, feedElement)
Edit a feed element.

updateFeedElementBookmarks(communityId, feedElementId, bookmarks)
Bookmark a feed element or remove a bookmark from a feed element using an input class.

updateFeedElementBookmarks(communityId, feedElementId, isBookmarkedByCurrentUser)
Bookmark a feed element or remove a bookmark from a feed element.

updateFeedElementReadByCapabilityBatch(communityId, feedElementIds, readBy)
Mark multiple feed elements as read by the context user at the same time using an input class.

updateFeedElementReadByCapabilityBatch(communityId, feedElementIds, isReadByMe)
Mark multiple feed elements as read by the context user at the same time.
updateLikeForComment(communityId, commentId, isLikedByCurrentUser)
Like or unlike a comment.

updateLikeForFeedElement(communityId, feedElementId, isLikedByCurrentUser)
Like or unlike a feed element.

updatePinnedFeedElements(communityId, feedType, subjectId, pin)
Pin or unpin feed elements to a group or topic feed.

updateStream(communityId, streamId, streamInput)
Update a Chatter feed stream.

voteOnFeedElementPoll(communityId, feedElementId, myChoiceId)
Vote on a poll or change your vote on a poll.

createStream(communityId, streamInput)
Create a Chatter feed stream.

API Version
39.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterStream createStream(String communityId,
ConnectApi.ChatterStreamInput streamInput)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

streamInput
Type: ConnectApi.ChatterStreamInput
A ConnectApi.ChatterStreamInput body.

Return Value
Type: ConnectApi.ChatterStream

deleteComment(communityId, commentId)
Delete a comment.
API Version
28.0

Requires Chatter
Yes

Signature
public static Void deleteComment(String communityId, String commentId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

commentId
Type: String
ID for a comment.

Return Value
Type: Void

deleteFeedElement(communityId, feedElementId)
Delete a feed element.

API Version
31.0

Requires Chatter
Yes

Signature
public static deleteFeedElement(String communityId, String feedElementId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element.
Return Value
Type: Void

**deleteLike(communityId, likeId)**
Delete a like on a comment or post.

**API Version**
28.0

**Requires Chatter**
Yes

**Signature**
```
public static Void deleteLike(String communityId, String likeId)
```

**Parameters**

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- **likeId**
  - Type: String
  - ID for a like.

**Return Value**
Type: Void

**deleteStream(communityId, streamId)**
Delete a Chatter feed stream.

**API Version**
39.0

**Requires Chatter**
Yes

**Signature**
```
public static Void deleteStream(String communityId, String streamId)
```
Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

streamId
Type: String
ID of the Chatter feed stream.

Return Value
Type: Void

getComment(communityId, commentId)
Get a comment.

API Version
28.0

Available to Guest Users
31.0

Requires Chatter
Yes

Signature
public static ConnectApi.Comment getComment(String communityId, String commentId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

commentId
Type: String
ID for a comment.

Return Value
Type: ConnectApi.Comment

getCommentBatch(communityId, commentIds)
Get a list of comments.
**API Version**

42.0

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.BatchResult[] getCommentBatch(String communityId, List<String> commentIds)
```

**Parameters**

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.
- `commentIds`
  - Type: `List<String>`
  - A list of up to 100 comment IDs.

**Return Value**

Type: `ConnectApi.BatchResult[]`

The `ConnectApi.BatchResult.getResult()` method returns a `ConnectApi.Comment` object and errors for comments that didn't load.

**getCommentInContext(communityId, commentId, pageSize)**

Get a threaded comment in the context of its parent comments and post.

**API Version**

44.0

**Available to Guest Users**

44.0

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.FeedElement getCommentInContext(String communityId, String commentId, Integer pageSize)
```
Parameters

*communityId*

- Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

*commentId*

- Type: `String`
  - ID of the comment.

*pageSize*

- Type: `Integer`
  - Specifies the number of items per page. Valid values are from 1 through 100. If you don't specify a value, the default size is 25.

Return Value

- Type: `ConnectApi.FeedElement`

If the comment doesn't support the `comments` capability, the return value is `ConnectApiNotFoundException`.

**getCommentsForFeedElement(communityId, feedElementId)**

Get comments for a feed element.

API Version

32.0

Available to Guest Users

32.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.CommentPage getCommentsForFeedElement(String communityId, String feedElementId)
```

Parameters

*communityId*

- Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

*feedElementId*

- Type: `String`
  - ID of the feed element.
Return Value
Type: `ConnectApi.CommentPage`
If the feed element doesn’t support the `Comments` capability, the return value is `ConnectApi.NotFoundException`.

`getCommentsForFeedElement(communityId, feedElementId, threadedCommentsCollapsed)`
Get comments in a threaded style for a feed element.

API Version
44.0

Available to Guest Users
44.0

Requires Chatter
Yes

Signature
```
public static ConnectApi.CommentPage getCommentsForFeedElement(String communityId,
                    String feedElementId, Boolean threadedCommentsCollapsed)
```

Parameters
```
communityId
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.

feedElementId
  Type: `String`
  ID of the feed element.

threadedCommentsCollapsed
  Type: `Boolean`
  Specifies whether to return threaded comments in a collapsed style (`true`) or not (`false`). If you pass in `null`, the default is `false`.
```

Return Value
Type: `ConnectApi.CommentPage`
If the feed element doesn’t support the `Comments` capability, the return value is `ConnectApi.NotFoundException`.

`getCommentsForFeedElement(communityId, feedElementId, pageParam, pageSize)`
Get a page of comments for a feed element.
API Version
32.0

Available to Guest Users
32.0

Requires Chatter
Yes

Signature
public static ConnectApi.CommentPage getCommentsForFeedElement(String communityId, String feedElementId, String pageParam, Integer pageSize)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of comments per page. Valid values are from 1 through 100. If you pass null, the default size is 25.

Return Value
Type: ConnectApi.CommentPage
If the feed element doesn’t support the Comments capability, the return value is ConnectApi.NotFoundException.

getCommentsForFeedElement(communityId, feedElementId, pageParam, pageSize, threadedCommentsCollapsed)
Get a page of comments in a threaded style for a feed element.

API Version
44.0
Available to Guest Users

44.0

Requires Chatter

Yes

Signature

public static ConnectApi.CommentPage getCommentsForFeedElement(String communityId,
String feedElementId, String pageParam, Integer pageSize, Boolean
threadedCommentsCollapsed)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of comments per page. Valid values are from 1 through 100. If you pass null, the default size is 25.

threadedCommentsCollapsed
Type: Boolean
Specifies whether to return threaded comments in a collapsed style (true) or not (false). If you pass in null, the default is false.

Return Value

Type: ConnectApi.CommentPage
If the feed element doesn’t support the Comments capability, the return value is ConnectApi.NotFoundException.

getCommentsForFeedElement(communityId, feedElementId,
threadedCommentsCollapsed, sortParam)

Get sorted comments in a threaded style for a feed element.

API Version

44.0
Available to Guest Users
44.0

Requires Chatter
Yes

Signature
public static ConnectApi.CommentsCapability getCommentsForFeedElement(String communityId, String feedElementId, Boolean threadedCommentsCollapsed, ConnectApi.FeedCommentSortOrder sortParam)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element.

threadedCommentsCollapsed
Type: Boolean
Specifies whether to return threaded comments in a collapsed style (true) or not (false). If you pass in null, the default is false.

sortParam
Type: ConnectApi.FeedCommentSortOrder
Order of comments. Values are:
• CreatedDateLatestAsc—Sorts by most recently created comments in ascending order.
• CreatedDateOldestAsc—Sorts by oldest comments in ascending order.
• Relevance—Sorts by most relevant content.
Sorting in descending order isn’t supported.

Return Value
Type: ConnectApi.CommentPage
If the feed element doesn’t support the Comments capability, the return value is ConnectApi.NotFoundException.

getCommentsForFeedElement(communityId, feedElementId, pageParam, pageSize, threadedCommentsCollapsed, sortParam)
Get a page of sorted comments in a threaded style for a feed element.
API Version
44.0

Available to Guest Users
44.0

Requires Chatter
Yes

Signature
public static ConnectApi.CommentPage getCommentsForFeedElement(String communityId,
String feedElementId, String pageParam, Integer pageSize, Boolean threadedCommentsCollapsed, ConnectApi.FeedCommentSortOrder sortParam)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of comments per page. Valid values are from 1 through 100. If you pass null, the default size is 25.

threadedCommentsCollapsed
Type: Boolean
Specifies whether to return threaded comments in a collapsed style (true) or not (false). If you pass in null, the default is false.

sortBy
Type: ConnectApi.FeedCommentSortOrder
Order of comments. Values are:
• CreatedDateLatestAsc—Sorts by most recently created comments in ascending order.
• CreatedDateOldestAsc—Sorts by oldest comments in ascending order.
• Relevance—Sorts by most relevant content.
Sorting in descending order isn’t supported.
Return Value
Type: ConnectApi.CommentPage
If the feed element doesn't support the Comments capability, the return value is ConnectApi.NotFoundException.

getCommentsForFeedElement(communityId, feedElementId, sortParam)
Get sorted comments for a feed element.

API Version
41.0

Available to Guest Users
41.0

Requires Chatter
Yes

Signature
public static ConnectApi.CommentsCapability getCommentsForFeedElement(String communityId, String feedElementId, ConnectApi.FeedCommentSortOrder sortParam)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element.

sortParam
Type: ConnectApi.FeedCommentSortOrder
Order of comments. Values are:
• CreatedDateLatestAsc—Sorts by most recently created comments in ascending order.
• CreatedDateOldestAsc—Sorts by oldest comments in ascending order.
• Relevance—Sorts by most relevant content.
  Sorting in descending order isn't supported.

Return Value
Type: ConnectApi.CommentsCapability
If the feed element doesn't support the Comments capability, the return value is ConnectApi.NotFoundException.
getCommentsForFeedElement(communityId, feedElementId, sortParam, threadedCommentsCollapsed)

Get sorted comments in a threaded style for a feed element.

API Version
44.0

Available to Guest Users
44.0

Requires Chatter
Yes

Signature
public static ConnectApi.CommentsCapability getCommentsForFeedElement(String communityId, String feedElementId, ConnectApi.FeedCommentSortOrder sortParam, Boolean threadedCommentsCollapsed)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element.

sortParam
Type: ConnectApi.FeedCommentSortOrder
Order of comments. Values are:
• CreatedDateLatestAsc—Sorts by most recently created comments in ascending order.
• CreatedDateOldestAsc—Sorts by oldest comments in ascending order.
• Relevance—Sorts by most relevant content.
Sorting in descending order isn’t supported.

threadedCommentsCollapsed
Type: Boolean
Specifies whether to return threaded comments in a collapsed style (true) or not (false). If you pass in null, the default is false.

Return Value
Type: ConnectApi.CommentsCapability
If the feed element doesn't support the Comments capability, the return value is `ConnectApi.NotFoundException`.

### getExtensions(communityId, pageParam, pageSize)

Get extensions.

**API Version**

40.0

**Available to Guest Users**

40.0

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.ExtensionDefinitions getExtensions(String communityId, String pageParam, Integer pageSize)
```

**Parameters**

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **pageParam**
  - Type: `String`
  - Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

- **pageSize**
  - Type: `Integer`
  - Specifies the number of items per page. Valid values are from 1 through 100. The default size is 15.

**Return Value**

Type: `ConnectApi.ExtensionDefinitions`

### getFeed(communityId, feedType)

Get a feed.

**API Version**

28.0
Available to Guest Users
32.0

Requires Chatter
Yes

Signature
public static ConnectApi.Feed getFeed(String communityId, ConnectApi.FeedType feedType)

Parameters
- communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.
- feedType
  Type: ConnectApi.FeedType
  Type of feed. Valid values are Company, DirectMessageModeration, DirectMessages, Home, Moderation, and PendingReview.

Return Value
Type: ConnectApi.Feed

getFeed(communityId, feedType, sortParam)
Get a sorted feed.

API Version
28.0

Available to Guest Users
32.0

Requires Chatter
Yes

Signature
public static ConnectApi.Feed getFeed(String communityId, ConnectApi.FeedType feedType, ConnectApi.FeedSortOrder sortParam)
Parameters

```
communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

feedType
  Type: ConnectApi.FeedType
  Type of feed. Valid values are Company, DirectMessageModeration, DirectMessages, Home, Moderation, and PendingReview.

sortParam
  Type: ConnectApi.FeedSortOrder
  Values are:
  • CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
  • CreatedDateDesc—Sorts by most recent creation date.
  • LastModifiedDateDesc—Sorts by most recent activity.
  • MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
  • Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

If you pass in null, the default value CreatedDateDesc is used.

If feedType is DirectMessages, sortParam must be LastModifiedDateDesc.
```

Return Value

```
Type: ConnectApi.Feed
```

getFeed(communityId, feedType, subjectId)

Get a feed for a record or user.

API Version

```
28.0
```

Available to Guest Users

```
32.0
```

Requires Chatter

```
Yes
```

Signature

```
public static ConnectApi.Feed getFeed(String communityId, ConnectApi.FeedType feedType, String subjectId)
```
Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Type of feed. Valid values include every ConnectApi.FeedType except Company, DirectMessageModeration, DirectMessages, Filter, Home, Landing, Moderation, and PendingReview.

subjectId
Type: String
If feedType is Record, subjectId can be any record ID, including a group ID. If feedType is Streams, subjectId must be a stream ID. If feedType is Topics, subjectId must be a topic ID. If feedType is UserProfile, subjectId can be any user ID. If the feedType is any other value, subjectId must be the ID of the context user or the alias me.

Return Value
Type: ConnectApi.Feed

getFeed(communityId, feedType, subjectId, sortParam)
Get a sorted feed for a record or user.

API Version
28.0

Available to Guest Users
32.0

Requires Chatter
Yes

Signature
public static ConnectApi.Feed getFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, ConnectApi.FeedSortOrder sortParam)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Type of feed. Valid values include every `ConnectApi.FeedType` except `Company`, `DirectMessageModeration`, `DirectMessages`, `Filter`, `Home`, `Landing`, `Moderation`, and `PendingReview`.

**subjectId**
Type: `String`
If `feedType` is `Record`, `subjectId` can be any record ID, including a group ID. If `feedType` is `Streams`, `subjectId` must be a stream ID. If `feedType` is `Topics`, `subjectId` must be a topic ID. If `feedType` is `UserProfile`, `subjectId` can be any user ID. If the `feedType` is any other value, `subjectId` must be the ID of the context user or the alias `me`.

**sortParam**
Type: `ConnectApi.FeedSortOrder`
Values are:
- `CreatedDateAsc`—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- `CreatedDateDesc`—Sorts by most recent creation date.
- `LastModifiedDateDesc`—Sorts by most recent activity.
- `MostViewed`—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- `Relevance`—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.

If you pass in `null`, the default value `CreatedDateDesc` is used.

**Return Value**
Type: `ConnectApi.Feed`

**getFeedDirectory(String)**
Get a list of all feeds available to the context user.

**API Version**
30.0

**Requires Chatter**
Yes

**Signature**
```java
public static ConnectApi.FeedDirectory getFeedDirectory(String communityId)
```

**Parameters**
- `communityId`
  Type: `String`
  ID for an Experience Cloud site, `internal`, or `null`. 
Return Value
Type: `ConnectApi.FeedDirectory`

`getFeedElement(communityId, feedElementId)`
Get a feed element.

API Version
31.0

Available to Guest Users
31.0

Requires Chatter
Yes

Signature
```java
public static ConnectApi.FeedElement getFeedElement(String communityId, String feedElementId)
```

Parameters

- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.

- `feedElementId`
  Type: `String`
  ID of the feed element.

Return Value
Type: `ConnectApi.FeedElement`

`getFeedElement(communityId, feedElementId, commentSort)`
Get a feed element with sorted comments.

API Version
41.0

Available to Guest Users
41.0
Requires Chatter
Yes

Signature

```java
public static ConnectApi.FeedElement getFeedElement(String communityId, String feedElementId, ConnectApi.FeedCommentSortOrder commentSort)
```

Parameters

- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.

- `feedElementId`
  Type: `String`
  ID of the feed element.

- `commentSort`
  Type: `ConnectApi.FeedCommentSortOrder`
  Order of comments.
  - CreatedDateLatestAsc—Sorts by most recently created comments in ascending order.
  - CreatedDateOldestAsc—Sorts by oldest comments in ascending order.
  - Relevance—Sorts by most relevant content.
  The default value is CreatedDateLatestAsc.
  Sorting in descending order isn’t supported.

Return Value

Type: `ConnectApi.FeedElement`

```java
getFeedElement(communityId, feedElementId, threadedCommentsCollapsed)
```

Get a feed element and its comments in a threaded style.

API Version

44.0

Available to Guest Users

44.0

Requires Chatter

Yes
Signature

```
public static ConnectApi.FeedElement getFeedElement(String communityId, String feedElementId, Boolean threadedCommentsCollapsed)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- `feedElementId`
  - Type: `String`
  - ID of the feed element.

- `threadedCommentsCollapsed`
  - Type: `Boolean`
  - Specifies whether to return threaded comments in a collapsed style (`true`) or not (`false`). If you pass in `null`, the default is `false`.

Return Value

Type: `ConnectApi.FeedElement`

```
getFeedElement(communityId, feedElementId, threadedCommentsCollapsed, commentSort)
```

Get a feed element and its sorted comments in a threaded style.

API Version

44.0

Available to Guest Users

44.0

Requires Chatter

Yes

Signature

```
public static ConnectApi.FeedElement getFeedElement(String communityId, String feedElementId, Boolean threadedCommentsCollapsed, ConnectApi.FeedCommentSortOrder commentSort)
```

Parameters

- `communityId`
  - Type: `String`
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element.

threadedCommentsCollapsed
Type: Boolean
Specifies whether to return threaded comments in a collapsed style (true) or not (false). If you pass in null, the default is false.

commentSort
Type: ConnectApi.FeedCommentSortOrder
Order of comments.
- CreatedDateLatestAsc—Sorts by most recently created comments in ascending order.
- CreatedDateOldestAsc—Sorts by oldest comments in ascending order.
- Relevance—Sorts by most relevant content.
  Sorting in descending order isn’t supported.

Return Value
Type: ConnectApi.FeedElement

getFeedElement(communityId, feedElementId, recentCommentCount, elementsPerBundle)
Get a feed element with the specified number of elements per bundle including no more than the specified number of comments per feed element.

API Version
31.0

Available to Guest Users
31.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedElement getFeedElement(String communityId, String feedElementId, Integer recentCommentCount, Integer elementsPerBundle)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

**feedElementId**
Type: String
ID of the feed element.

**recentCommentCount**
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

**elementsPerBundle**
Type: Integer
Maximum number of feed elements per bundle. The default and maximum value is 10.

Return Value
Type: `ConnectApi.FeedElement`

```java
getFeedElement(communityId, feedElementId, recentCommentCount, elementsPerBundle, threadedCommentsCollapsed)
```
Get a feed element with its comments in a threaded style with the specified number of elements per bundle and comments per feed element.

API Version
44.0

Available to Guest Users
44.0

Requires Chatter
Yes

Signature
```java
public static ConnectApi.FeedElement getFeedElement(String communityId, String feedElementId, Integer recentCommentCount, Integer elementsPerBundle, Boolean threadedCommentsCollapsed)
```

**Parameters**

**communityId**
Type: String
ID for an Experience Cloud site, internal, or null.

**feedElementId**
Type: String
ID of the feed element.
recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

elementsPerBundle
Type: Integer
Maximum number of feed elements per bundle. The default and maximum value is 10.
threadedCommentsCollapsed
Type: Boolean
Specifies whether to return threaded comments in a collapsed style (true) or not (false). If you pass in null, the default is false.

Return Value
Type: ConnectApi.FeedElement

getFeedElement(communityId, feedElementId, recentCommentCount, elementsPerBundle, threadedCommentsCollapsed, commentSort)
Get a feed element with its sorted comments in a threaded style with the specified number of elements per bundle and comments per feed element.

API Version
44.0

Available to Guest Users
44.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedElement getFeedElement(String communityId, String feedElementId, Integer recentCommentCount, Integer elementsPerBundle, Boolean threadedCommentsCollapsed, ConnectApi.FeedCommentSortOrder commentSort)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element.
recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

elementsPerBundle
Type: Integer
Maximum number of feed elements per bundle. The default and maximum value is 10.
threadedCommentsCollapsed
Type: Boolean
Specifies whether to return threaded comments in a collapsed style (true) or not (false). If you pass in null, the default is false.

commentSort
Type: ConnectApi.FeedCommentSortOrder
Order of comments.
  • CreatedDateLatestAsc—Sorts by most recently created comments in ascending order.
  • CreatedDateOldestAsc—Sorts by oldest comments in ascending order.
  • Relevance—Sorts by most relevant content.
  Sorting in descending order isn’t supported.

Return Value
Type: ConnectApi.FeedElement

getFeedElement(communityId, feedElementId, recentCommentCount, elementsPerBundle, commentSort)
Get a feed element with the specified number of elements per bundle including no more than the specified number of sorted comments per feed element.

API Version
41.0

Available to Guest Users
41.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedElement getFeedElement(String communityId, String feedElementId, Integer recentCommentCount, Integer elementsPerBundle, ConnectApi.FeedCommentSortOrder commentSort)
Parameters

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- **feedElementId**
  - Type: String
  - ID of the feed element.

- **recentCommentCount**
  - Type: Integer
  - Maximum number of comments to return with each feed element. The default value is 3.

- **elementsPerBundle**
  - Type: Integer
  - Maximum number of feed elements per bundle. The default and maximum value is 10.

- **commentSort**
  - Type: ConnectApi.FeedCommentSortOrder
    - Order of comments.
    - **CreatedDateLatestAsc**—Sorts by most recently created comments in ascending order.
    - **CreatedDateOldestAsc**—Sorts by oldest comments in ascending order.
    - **Relevance**—Sorts by most relevant content.
    - The default value is CreatedDateLatestAsc.
    - Sorting in descending order isn’t supported.

Return Value

- Type: ConnectApi.FeedElement

**getFeedElementBatch(communityId, feedElementIds)**

Get a list of feed elements.

API Version

- 31.0

Available to Guest Users

- 32.0

Requires Chatter

- Yes
Signature

public static ConnectApi.BatchResult[] getFeedElementBatch(String communityId, List<String> feedElementIds)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementIds
Type: List<String>
A list of up to 500 feed element IDs.

Return Value

Type: ConnectApi.BatchResult[]
The ConnectApi.BatchResult.getResult() method returns a ConnectApi.FeedElement object and errors for feed elements that didn’t load.

getFeedElementPoll(communityId, feedElementId)
Get the poll associated with a feed element.

API Version

32.0

Available to Guest Users

32.0

Requires Chatter

Yes

Signature

public static ConnectApi.PollCapability getFeedElementPoll(String communityId, String feedElementId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element.
Return Value

Type: `ConnectApi.PollCapability`

If the feed element doesn’t support this capability, the return value is `ConnectApi.NotFoundException`.

Note: Triggers on FeedItem objects run before their attachment and capabilities information is saved, which means that
`ConnectApi.FeedItem.attachment` information and `ConnectApi.FeedElement.capabilities` information
may not be available in the trigger.

`getFeedElementsFromBundle(communityId, feedElementId)`

Get feed elements from a bundle.

API Version

31.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.FeedElementPage getFeedElementsFromBundle(String communityId, String feedElementId)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, `internal`, or `null`.

- `feedElementId`
  - Type: `String`
  - ID of the feed element.

Return Value

Type: `ConnectApi.FeedElementPage`

`getFeedElementsFromBundle(communityId, feedElementId, pageParam, pageSize, elementsPerBundle, recentCommentCount)`

Get a page of feed elements from a bundle. Specify the number of elements per bundle and include no more than the specified number
of comments per feed element.

API Version

31.0
Requires Chatter
Yes

Signature
public static ConnectApi.FeedElementPage getFeedElementsFromBundle(String communityId,
String feedElementId, String pageParam, Integer pageSize, Integer elementsPerBundle,
Integer recentCommentCount)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element.

pageParam
Type: String
Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

elementsPerBundle
Type: Integer
Maximum number of feed elements per bundle. The default and maximum value is 10.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

Return Value
Type: ConnectApi.FeedElementPage

getFeedElementsFromFeed(communityId, feedType)

API Version
31.0
Available to Guest Users
31.0

Requires Chatter
Yes

Signature

```java
public static ConnectApi.FeedElementPage getFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType)
```

Parameters

`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

`feedType`
Type: `ConnectApi.FeedType`
Type of feed. Valid values are `Company`, `DirectMessageModeration`, `DirectMessages`, `Home`, `Moderation`, and `PendingReview`.

Return Value
Type: `ConnectApi.FeedElementPage`

Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

```java
setTestGetFeedElementsFromFeed(communityId, feedType, result)
```


**getFeedElementsFromFeed(communityId, feedType, pageParam, pageSize, sortParam)**


API Version
31.0

Available to Guest Users
31.0
Requires Chatter

Yes

Signature

```java
public static ConnectApi.FeedElementPage getFeedElementsFromFeed(String communityId,
ConnectApi.FeedType feedType, String pageParam, Integer pageSize,
ConnectApi.FeedSortOrder sortParam)
```

Parameters

**communityId**

Type: `String`

ID for an Experience Cloud site, `internal`, or `null`.

**feedType**

Type: `ConnectApi.FeedType`

Type of feed. Valid values are `Company`, `DirectMessageModeration`, `DirectMessages`, `Home`, `Moderation`, and `PendingReview`.

**pageParam**

Type: `String`

Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**

Type: `Integer`

Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**

Type: `ConnectApi.FeedSortOrder`

Values are:

- `CreatedDateAsc`—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- `CreatedDateDesc`—Sorts by most recent creation date.
- `LastModifiedDateDesc`—Sorts by most recent activity.
- `MostViewed`—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- `Relevance`—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.

If you pass in `null`, the default value `CreatedDateDesc` is used.

If `feedType` is `DirectMessages`, `sortParam` must be `LastModifiedDesc`.

Return Value

Type: `ConnectApi.FeedElementPage`
Usage

To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- setTestGetFeedElementsFromFeed(communityId, feedType, pageParam, pageSize, sortParam, result)
  
  *Apex Developer Guide: Testing ConnectApi Code*

### getFeedElementsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam)

Get a page of sorted feed elements from the Company, DirectMessageModeration, DirectMessages, Home, Moderation, and PendingReview feeds. Each feed element contains no more than the specified number of comments.

API Version

31.0

Available to Guest Users

31.0

Requires Chatter

Yes

### Signature

`public static ConnectApi.FeedElementPage getFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam)`

### Parameters

- **communityId**
  
  Type: `String`

  ID for an Experience Cloud site, internal, or null.

- **feedType**
  
  Type: `ConnectApi.FeedType`

  Type of feed. Valid values are Company, DirectMessageModeration, DirectMessages, Home, Moderation, and PendingReview.

- **recentCommentCount**
  
  Type: `Integer`

  Maximum number of comments to return with each feed item. The default value is 3.

- **density**
  
  Type: `ConnectApi.FeedDensity`
Specify the amount of content in a feed.

- **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

`pageParam`

Type: `String`  
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

`pageSize`

Type: `Integer`  
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

`sortParam`

Type: `ConnectApi.FeedSortOrder`  
Values are:

- **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifiedDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- **Relevance**—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.

If you pass in `null`, the default value `CreatedDateDesc` is used.

If `feedType` is `DirectMessages`, `sortParam` must be `LastModifiedDesc`.

**Return Value**

Type: `ConnectApi.FeedElementPage`

**Usage**

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

`setTestGetFeedElementsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, result)`  
*Apex Developer Guide: Testing ConnectApi Code*

`getFeedElementsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, filter)`  
Get a page of sorted and filtered feed elements from the `Home` feed. Each feed element contains no more than the specified number of comments.
API Version

32.0

Available to Guest Users

32.0

Requires Chatter

Yes

Signature

public static ConnectApi.FeedElementPage getFeedElementsFromFeed(String communityId,
ConnectApi.FeedType feedType, Integer recentCommentCount, ConnectApi.FeedDensity density,
String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam,
ConnectApi.FeedFilter filter)

Parameters

communityId

Type: String

ID for an Experience Cloud site, internal, or null.

feedType

Type: ConnectApi.FeedType

The type of feed. The only valid value is Home.

recentCommentCount

Type: Integer

Maximum number of comments to return with each feed item. The default value is 3.

density

Type: ConnectApi.FeedDensity

Specify the amount of content in a feed.

• AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.

• FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam

Type: String

Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

When the sortParam is MostViewed, you must pass in null for the pageParam.

pageSize

Type: Integer

Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.
When the `sortParam` is `MostViewed`, the `pageSize` must be a value from 1 to 25.

`sortParam`
Type: `ConnectApi.FeedSortOrder`
Values are:
- `CreatedDateAsc`—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- `CreatedDateDesc`—Sorts by most recent creation date.
- `LastModifiedDateDesc`—Sorts by most recent activity.
- `MostViewed`—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- `Relevance`—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.

If you pass in `null`, the default value `CreatedDateDesc` is used.

`filter`
Type: `ConnectApi.FeedFilter`
Specifies the feed filters.

- `AllQuestions`—Feed elements that are questions.
- `AuthoredBy`—Feed elements authored by the user profile owner. This value is valid only for the `UserProfile` feed.
- `CommunityScoped`—Feed elements that are scoped to Experience Cloud sites. Currently, these feed elements have a User or a Group parent record. However, other parent record types could be scoped to sites in the future. Feed elements that are always visible in all sites are filtered out. This value is valid only for the `UserProfile` feed.
- `QuestionsWithCandidateAnswers`—Feed elements that are questions that have candidate answers associated with them. This value is valid only for users with the `Access Einstein-Generated Answers` permission.
- `QuestionsWithCandidateAnswersReviewedPublished`—Feed elements that are questions that have candidate answers that have been reviewed or published. This value is valid only for users with the `Access Einstein-Generated Answers` permission.
- `Read`—Feed elements that are older than 30 days or are marked as read for the context user. Includes existing feed elements when the context user joined the group. This value is valid only for the `Record` feed of a group.
- `SolvedQuestions`—Feed elements that are questions and that have a best answer.
- `UnansweredQuestions`—Feed elements that are questions and that don't have any answers.
- `UnansweredQuestionsWithCandidateAnswers`—Feed elements that are questions that don't have answers but have candidate answers associated with them. This value is valid only for users with the `Access Einstein-Generated Answers` permission.
- `Unread`—Feed elements that are created in the past 30 days and aren't marked as read for the context user. This value is valid only for the `Record` feed of a group.
- `UnsolvedQuestions`—Feed elements that are questions and that don't have a best answer.

**Return Value**
Type: `ConnectApi.FeedElementPage`
Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetFeedElementsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, filter, result)


getFeedElementsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, filter, threadedCommentsCollapsed)
Get a page of filtered and sorted feed elements with comments in a threaded style from the Home feed. Each feed element contains no more than the specified number of comments.

API Version
44.0

Available to Guest Users
44.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedElementPage getFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedFilter filter, Boolean threadedCommentsCollapsed)

Parameters

  communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

  feedType
  Type: ConnectApi.FeedType
  The type of feed. The only valid value is Home.

  recentCommentCount
  Type: Integer
  Maximum number of comments to return with each feed item. The default value is 3.
density
Type: `ConnectApi.FeedDensity`
Specify the amount of content in a feed.

- **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

`pageParam`
Type: `String`
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

When the `sortParam` is `MostViewed`, you must pass in `null` for the `pageParam`.

`pageSize`
Type: `Integer`
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

When the `sortParam` is `MostViewed`, the `pageSize` must be a value from 1 to 25.

`sortParam`
Type: `ConnectApi.FeedSortOrder`
Values are:

- **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- **Relevance**—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.

If you pass in `null`, the default value `CreatedDateDesc` is used.

`filter`
Type: `ConnectApi.FeedFilter`
Specifies the feed filters.

- **AllQuestions**—Feed elements that are questions.
- **AuthoredBy**—Feed elements authored by the user profile owner. This value is valid only for the `UserProfile` feed.
- **CommunityScoped**—Feed elements that are scoped to Experience Cloud sites. Currently, these feed elements have a User or a Group parent record. However, other parent record types could be scoped to sites in the future. Feed elements that are always visible in all sites are filtered out. This value is valid only for the `UserProfile` feed.
- **QuestionsWithCandidateAnswers**—Feed elements that are questions that have candidate answers associated with them. This value is valid only for users with the `Access Einstein-Generated Answers` permission.
- **QuestionsWithCandidateAnswersReviewedPublished**—Feed elements that are questions that have candidate answers that have been reviewed or published. This value is valid only for users with the `Access Einstein-Generated Answers` permission.
- **Read**—Feed elements that are older than 30 days or are marked as read for the context user. Includes existing feed elements when the context user joined the group. This value is valid only for the **Record** feed of a group.
- **SolvedQuestions**—Feed elements that are questions and that have a best answer.
- **UnansweredQuestions**—Feed elements that are questions and that don’t have any answers.
- **UnansweredQuestionsWithCandidateAnswers**—Feed elements that are questions that don’t have answers but have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **Unread**—Feed elements that are created in the past 30 days and aren’t marked as read for the context user. This value is valid only for the **Record** feed of a group.
- **UnsolvedQuestions**—Feed elements that are questions and that don’t have a best answer.

**threadedCommentsCollapsed**
Type: **Boolean**
Specifies whether to return threaded comments in a collapsed style (**true**) or not (**false**). If you pass in **null**, the default is **false**.

**Return Value**
Type: **ConnectApi.FeedElementPage**

**Usage**
To test code that uses this method, use the matching set test method (prefix the method name with **setTest**). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- **setTestGetFeedElementsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, filter, threadedCommentsCollapsed, result)**
  *Apex Developer Guide: Testing ConnectApi Code*

**getFeedElementsFromFeed(communityId, feedType, subjectId)**
Get feed elements from any feed other than **Company**, **DirectMessageModeration**, **DirectMessages**, **Filter**, **Home**, **Landing**, **Moderation**, and **PendingReview** for a user or record.

**API Version**
31.0

**Available to Guest Users**
31.0

**Requires Chatter**
Yes
Signature

```java
public static ConnectApi.FeedElementPage getFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId)
```

Parameters

- `communityId`
  - **Type:** String
  - ID for an Experience Cloud site, internal, or null.

- `feedType`
  - **Type:** ConnectApi.FeedType
  - Type of feed. Valid values include every ConnectApi.FeedType except Company, DirectMessageModeration, DirectMessages, Filter, Home, Landing, Moderation, and PendingReview.

- `subjectId`
  - **Type:** String
  - If `feedType` is Record, `subjectId` can be any record ID, including a group ID. If `feedType` is Streams, `subjectId` must be a stream ID. If `feedType` is Topics, `subjectId` must be a topic ID. If `feedType` is UserProfile, `subjectId` can be any user ID. If the `feedType` is any other value, `subjectId` must be the ID of the context user or the alias me.

Return Value

- **Type:** ConnectApi.FeedElementPage

Usage

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

Example for Getting the Context User’s News Feed

```java
ConnectApi.FeedElementPage fep =
ConnectApi.ChatterFeeds.getFeedElementsFromFeed(Network.getNetworkId(),
ConnectApi.FeedType.News, 'me');
```

Example for Getting Another User’s Profile Feed

```java
ConnectApi.FeedElementPage fep =
ConnectApi.ChatterFeeds.getFeedElementsFromFeed(Network.getNetworkId(),
ConnectApi.FeedType.UserProfile, '005R0000000HwMA');
```
Example for Getting Another User’s Record Feed

```java
ConnectApi.FeedElementPage fep =
ConnectApi.ChatterFeeds.getFeedElementsFromFeed(Network.getNetworkId(),
ConnectApi.FeedType.Record, '005R0000000HwMA');
```

SEE ALSO:

```
setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, result)

```

**getFeedElementsFromFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam)**

Get a page of sorted feed elements from any feed other than Company, DirectMessageModeration, DirectMessages, Filter, Home, Landing, Moderation, and PendingReview.

**API Version**

31.0

**Available to Guest Users**

31.0

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.FeedElementPage getFeedElementsFromFeed(String communityId,
ConnectApi.FeedType feedType, String subjectId, String pageParam, Integer pageSize,
ConnectApi.FeedSortOrder sortParam)
```

**Parameters**

- **communityId**
  - **Type:** String
  - ID for an Experience Cloud site, internal, or null.

- **feedType**
  - **Type:** ConnectApi.FeedType
  - Type of feed. Valid values include every ConnectApi.FeedType except Company, DirectMessageModeration, DirectMessages, Filter, Home, Landing, Moderation, and PendingReview.

- **subjectId**
  - **Type:** String
  - If `feedType` is Record, `subjectId` can be any record ID, including a group ID. If `feedType` is Streams, `subjectId` must be a stream ID. If `feedType` is Topics, `subjectId` must be a topic ID. If `feedType` is UserProfile, `subjectId` can be any user ID. If the `feedType` is any other value, `subjectId` must be the ID of the context user or the alias me.
**getPageParam**

Type: `String`

Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**

Type: `Integer`

The number of feed elements per page.

**sortParam**

Type: `ConnectApi.FeedSortOrder`

Values are:
- `CreatedDateAsc`—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- `CreatedDateDesc`—Sorts by most recent creation date.
- `LastModifiedDateDesc`—Sorts by most recent activity.
- `MostViewed`—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- `Relevance`—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.

If you pass in `null`, the default value `CreatedDateDesc` is used.

**Return Value**

Type: `ConnectApi.FeedElementPage`

**Usage**

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

**SEE ALSO:**

`setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam, result)`

*Apex Developer Guide: Testing ConnectApi Code*

**getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam)**

Get a page of sorted feed elements from any feed other than `Company`, `DirectMessageModeration`, `DirectMessages`, `Filter`, `Home`, `Landing`, `Moderation`, and `PendingReview`. Each feed element includes no more than the specified number of comments.

**API Version**

31.0
Available to Guest Users

31.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.FeedElementPage getFeedElementsFromFeed(String communityId,
ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount,
ConnectApi.FeedDensity density, String pageParam, Integer pageSize,
ConnectApi.FeedSortOrder sortParam)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, `internal`, or `null`.

- `feedType`
  - Type: `ConnectApi.FeedType`
  - Type of feed. Valid values include every `ConnectApi.FeedType` except `Company`, `DirectMessageModeration`, `DirectMessages`, `Filter`, `Home`, `Landing`, `Moderation`, and `PendingReview`.

- `subjectId`
  - Type: `String`
  - If `feedType` is `Record`, `subjectId` can be any record ID, including a group ID. If `feedType` is `Streams`, `subjectId` must be a stream ID. If `feedType` is `Topics`, `subjectId` must be a topic ID. If `feedType` is `UserProfile`, `subjectId` can be any user ID. If the `feedType` is any other value, `subjectId` must be the ID of the context user or the alias `me`.

- `recentCommentCount`
  - Type: `Integer`
  - Maximum number of comments to return with each feed element. The default value is 3.

- `density`
  - Type: `ConnectApi.FeedDensity`
  - Specify the amount of content in a feed.
    - *AllUpdates*—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
    - *FewerUpdates*—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

- `pageParam`
  - Type: `String`
  - Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

- `pageSize`
  - Type: `Integer`
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**

Type: `ConnectApi.FeedSortOrder`

Values are:

- `CreatedDateAsc`—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- `CreatedDateDesc`—Sorts by most recent creation date.
- `LastModifiedDateDesc`—Sorts by most recent activity.
- `MostViewed`—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- `Relevance`—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.

If you pass in `null`, the default value `CreatedDateDesc` is used.

**Return Value**

Type: `ConnectApi.FeedElementPage`

**Usage**

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

**SEE ALSO:**

- `setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, result)`
  - *Apex Developer Guide: Testing ConnectApi Code*

### getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, showInternalOnly)

Get a page of sorted feed elements from a record feed. Each feed element includes no more than the specified number of comments. Specify whether to return feed elements posted by internal (non-Experience Cloud site) users only.

**API Version**

31.0

**Available to Guest Users**

31.0

**Requires Chatter**

Yes
public static ConnectApi.FeedElementPage getFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, Boolean showInternalOnly)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Value must be ConnectApi.FeedType.Record.

subjectId
Type: String
Any record ID, including a group ID.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed item. The default value is 3.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
• MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.

• Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

If you pass in null, the default value CreatedDateDesc is used.

showInternalOnly
Type: Boolean
Specifies whether to show only feed items from internal (non-Experience Cloud site) users (true), or not (false). The default value is false.

Return Value
Type: ConnectApi.FeedElementPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, showInternalOnly, result)


getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, filter)
Get a page of sorted and filtered feed elements from the UserProfile feed.

API Version
35.0

Available to Guest Users
35.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedElementPage getFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedFilter filter)
Parameters

**communityId**
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

**feedType**
Type: `ConnectApi.FeedType`
Value must be `ConnectApi.FeedType.UserProfile`.

**subjectId**
Type: `String`
ID of any user. To specify the context user, use the user ID or the alias `me`.

**recentCommentCount**
Type: `Integer`
Maximum number of comments to return with each feed element. The default value is 3.

**density**
Type: `ConnectApi.FeedDensity`
Specify the amount of content in a feed.
- **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**
Type: `String`
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**
Type: `Integer`
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**
Type: `ConnectApi.FeedSortOrder`
Values are:
- **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- **Relevance**—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.
If you pass in `null`, the default value `CreatedDateDesc` is used.

**filter**
Type: `ConnectApi.FeedFilter`
Value must be ConnectApi.FeedFilter.CommunityScoped or ConnectApi.FeedFilter.AuthoredBy.

Return Value
Type: ConnectApi.FeedElementPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

Example
This example gets only community-specific feed elements.

```java
ConnectApi.FeedElementPage fep = ConnectApi.ChatterFeeds.getFeedElementsFromFeed(Network.getNetworkId(),
ConnectApi.FeedType.UserProfile, 'me', 3, ConnectApi.FeedDensity.FewerUpdates, null, null,
ConnectApi.FeedSortOrder.LastModifiedDateDesc, ConnectApi.FeedFilter.CommunityScoped);
```

SEE ALSO:
setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize,
sortParam, filter, result)

getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, filter, threadedCommentsCollapsed)
Get a page of feed elements with comments in a threaded style from the UserProfile feed.

API Version
44.0

Available to Guest Users
44.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedElementPage getFeedElementsFromFeed(String communityId,
ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount,
ConnectApi.FeedDensity density, String pageParam, Integer pageSize,
ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedFilter filter, Boolean threadedCommentsCollapsed)
Parameters

`communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.

`feedType`
  Type: `ConnectApi.FeedType`
  Value must be `ConnectApi.FeedType.UserProfile`.

`subjectId`
  Type: `String`
  ID of any user. To specify the context user, use the user ID or the alias `me`.

`recentCommentCount`
  Type: `Integer`
  Maximum number of comments to return with each feed element. The default value is 3.

`density`
  Type: `ConnectApi.FeedDensity`
  Specify the amount of content in a feed.
  - AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
  - FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

`pageParam`
  Type: `String`
  Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

`pageSize`
  Type: `Integer`
  Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

`sortParam`
  Type: `ConnectApi.FeedSortOrder`
  Values are:
  - CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
  - CreatedDateDesc—Sorts by most recent creation date.
  - LastModifiedDesc—Sorts by most recent activity.
  - MostViewed—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
  - Relevance—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.
  If you pass in `null`, the default value `CreatedDateDesc` is used.

`filter`
  Type: `ConnectApi.FeedFilter`
Value must be `ConnectApi.FeedFilter.CommunityScoped` or `ConnectApi.FeedFilter.AuthoredBy`.

`threadedCommentsCollapsed`
- **Type:** Boolean
- Specifies whether to return threaded comments in a collapsed style (`true`) or not (`false`). If you pass in `null`, the default is `false`.

**Return Value**
- **Type:** `ConnectApi.FeedElementPage`

**Usage**
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

**SEE ALSO:**
- `setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, filter, threadedCommentsCollapsed, result)`

```java
getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, customFilter)
```

Get a page of sorted and filtered feed elements from the case feed.

**API Version**
40.0

**Available to Guest Users**
40.0

**Requires Chatter**
Yes

**Signature**
```java
public static ConnectApi.FeedElementPage getFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String customFilter)
```

**Parameters**
- `communityId`
  - **Type:** String
ID for an Experience Cloud site, internal, or null.

**feedType**
- Type: `ConnectApi.FeedType`
- Value must be `ConnectApi.FeedType.Record`.

**subjectId**
- Type: `String`
- The ID of a case.

**recentCommentCount**
- Type: `Integer`
- Maximum number of comments to return with each feed element. The default value is 3.

**density**
- Type: `ConnectApi.FeedDensity`
- Specify the amount of content in a feed.
  - `AllUpdates`—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
  - `FewerUpdates`—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**
- Type: `String`
- Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**
- Type: `Integer`
- Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**
- Type: `ConnectApi.FeedSortOrder`
- Values are:
  - `CreatedDateAsc`—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
  - `CreatedDateDesc`—Sorts by most recent creation date.
  - `LastModifiedDateDesc`—Sorts by most recent activity.
  - `MostViewed`—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
  - `Relevance`—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.
- If you pass in `null`, the default value `CreateDateDesc` is used.

**customFilter**
- Type: `String`
- Custom filter that applies only to the case feed. See `customFeedFilter` in the Metadata API Developer Guide for supported values.
Return Value

Type: `ConnectApi.FeedElementPage`

Usage

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- `setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, customFilter, result)`

`getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly)`

Get a page of sorted feed elements from a record feed. Specify the number of elements per bundle and include no more than the specified number of comments per feed element. Specify whether to return feed elements posted by internal (non-Experience Cloud site) users only.

API Version

31.0

Available to Guest Users

31.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.FeedElementPage getFeedElementsFromFeed(String communityId,
ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, Integer
elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize,
ConnectApi.FeedSortOrder sortParam, Boolean showInternalOnly)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or null.

- `feedType`
  - Type: `ConnectApi.FeedType`
  - Value must be `ConnectApi.FeedType.Record`.
subjectId
Type: String
Any record ID, including a group ID.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed item. The default value is 3.

elementsPerBundle
Type: Integer
Maximum number of feed elements per bundle. The default and maximum value is 10.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

If you pass in null, the default value CreatedDateDesc is used.

showInternalOnly
Type: Boolean
Specifies whether to show only feed items from internal (non-Experience Cloud site) users (true), or not (false). The default value is false.
Return Value

Type: ConnectApi.FeedElementPage

Usage

To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly, result)

getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly, filter)

Get a page of sorted and filtered feed elements from a record feed. Specify the number of elements per bundle and include no more than the specified number of comments per feed element. Specify whether to return feed elements posted by internal (non-Experience Cloud site) users only.

API Version

32.0

Available to Guest Users

32.0

Requires Chatter

Yes

Signature

public static ConnectApi.FeedElementPage getFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, Integer elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, Boolean showInternalOnly, ConnectApi.FeedFilter filter)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Value must be `ConnectApi.FeedType.Record`.

**subjectId**

*Type: String*

Any record ID, including a group ID.

**recentCommentCount**

*Type: Integer*

Maximum number of comments to return with each feed item. The default value is 3.

**elementsPerBundle**

*Type: Integer*

Maximum number of feed elements per bundle. The default and maximum value is 10.

**density**

*Type: `ConnectApi.FeedDensity`*

Specify the amount of content in a feed.

- **AllUpdates**— Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**— Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**

*Type: String*

Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**

*Type: Integer*

Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**

*Type: `ConnectApi.FeedSortOrder`*

Values are:

- **CreatedDateAsc**— Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- **CreatedDateDesc**— Sorts by most recent creation date.
- **LastModifiedDateDesc**— Sorts by most recent activity.
- **MostViewed**— Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- **Relevance**— Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.

If you pass in `null`, the default value `ModifiedDateDesc` is used.

**showInternalOnly**

*Type: Boolean*

Specifies whether to show only feed items from internal (non-Experience Cloud site) users (`true`), or not (`false`). The default value is `false`.

**filter**

*Type: `ConnectApi.FeedFilter`*
Specifies the feed filters.

- **AllQuestions**—Feed elements that are questions.
- **AuthoredBy**—Feed elements authored by the user profile owner. This value is valid only for the *UserProfile* feed.
- **CommunityScoped**—Feed elements that are scoped to Experience Cloud sites. Currently, these feed elements have a User or a Group parent record. However, other parent record types could be scoped to sites in the future. Feed elements that are always visible in all sites are filtered out. This value is valid only for the *UserProfile* feed.
- **QuestionsWithCandidateAnswers**—Feed elements that are questions that have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **QuestionsWithCandidateAnswersReviewedPublished**—Feed elements that are questions that have candidate answers that have been reviewed or published. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **Read**—Feed elements that are older than 30 days or are marked as read for the context user. Includes existing feed elements when the context user joined the group. This value is valid only for the *Record* feed of a group.
- **SolvedQuestions**—Feed elements that are questions and that have a best answer.
- **UnansweredQuestions**—Feed elements that are questions and that don’t have any answers.
- **UnansweredQuestionsWithCandidateAnswers**—Feed elements that are questions that don’t have answers but have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **Unread**—Feed elements that are created in the past 30 days and aren’t marked as read for the context user. This value is valid only for the *Record* feed of a group.
- **UnsolvedQuestions**—Feed elements that are questions and that don’t have a best answer.

**Return Value**

Type: `ConnectApi.FeedElementPage`

**Usage**

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

**SEE ALSO:**

- `setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly, filter, result)`

```java
getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly, filter, threadedCommentsCollapsed)
```

Get a page of sorted and filtered feed elements with comments in a threaded style for a record feed. Specify the number of elements per bundle and include no more than the specified number of comments per feed element. Specify whether to return feed elements posted by internal (non-Experience Cloud site) users only.
API Version
44.0

Available to Guest Users
44.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedElementPage getFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, Integer elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, Boolean showInternalOnly, ConnectApi.FeedFilter filter, Boolean threadedCommentsCollapsed)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Value must be ConnectApi.FeedType.Record.

subjectId
Type: String
Any record ID, including a group ID.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed item. The default value is 3.

elementsPerBundle
Type: Integer
Maximum number of feed elements per bundle. The default and maximum value is 10.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.

- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**
Type: `Integer`
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**
Type: `ConnectApi.FeedSortOrder`
Values are:
- `CreatedDateAsc`—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- `CreatedDateDesc`—Sorts by most recent creation date.
- `LastModifiedDateDesc`—Sorts by most recent activity.
- `MostViewed`—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- `Relevance`—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.

If you pass in `null`, the default value `CreatedDateDesc` is used.

**showInternalOnly**
Type: `Boolean`
Specifies whether to show only feed items from internal (non-Experience Cloud site) users (true), or not (false). The default value is `false`.

**filter**
Type: `ConnectApi.FeedFilter`
Specifies the feed filters.
- `AllQuestions`—Feed elements that are questions.
- `AuthoredBy`—Feed elements authored by the user profile owner. This value is valid only for the `UserProfile` feed.
- `CommunityScoped`—Feed elements that are scoped to Experience Cloud sites. Currently, these feed elements have a User or a Group parent record. However, other parent record types could be scoped to sites in the future. Feed elements that are always visible in all sites are filtered out. This value is valid only for the `UserProfile` feed.
- `QuestionsWithCandidateAnswers`—Feed elements that are questions that have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
- `QuestionsWithCandidateAnswersReviewedPublished`—Feed elements that are questions that have candidate answers that have been reviewed or published. This value is valid only for users with the Access Einstein-Generated Answers permission.
- `Read`—Feed elements that are older than 30 days or are marked as read for the context user. Includes existing feed elements when the context user joined the group. This value is valid only for the `Record` feed of a group.
- `SolvedQuestions`—Feed elements that are questions and that have a best answer.
- `UnansweredQuestions`—Feed elements that are questions and that don’t have any answers.
- `UnansweredQuestionsWithCandidateAnswers`—Feed elements that are questions that don’t have answers but have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
- `Unread`—Feed elements that are created in the past 30 days and aren’t marked as read for the context user. This value is valid only for the `Record` feed of a group.
UnsolvedQuestions—Feed elements that are questions and that don't have a best answer.

threadedCommentsCollapsed
Type: Boolean
Specifies whether to return threaded comments in a collapsed style (true) or not (false). If you pass in null, the default is false.

Return Value
Type: ConnectApi.FeedElementPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly, filter, threadedCommentsCollapsed, result)


getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly, customFilter)
Get a page of sorted and filtered feed elements from a case feed.

API Version
40.0

Available to Guest Users
40.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedElementPage getFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, Integer elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, Boolean showInternalOnly, String customFilter)
Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Value must be ConnectApi.FeedType.Record.

subjectId
Type: String
The ID of a case.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed item. The default value is 3.

elementsPerBundle
Type: Integer
Maximum number of feed elements per bundle. The default and maximum value is 10.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
  • AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
  • FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

textParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
  • CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
  • CreatedDateDesc—Sorts by most recent creation date.
  • LastModifiedDateDesc—Sorts by most recent activity.
  • MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
  • Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
If you pass in null, the default value CreatedDateDesc is used.

**showInternalOnly**

Type: Boolean

Specifies whether to show only feed items from internal (non-Experience Cloud site) users (true), or not (false). The default value is false.

**customFilter**

Type: String

Custom filter that applies only to the case feed. See customFeedFilter in the Metadata API Developer Guide for supported values.

**Return Value**

Type: ConnectApi.FeedElementPage

**Usage**

To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

**SEE ALSO:**

setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly, customFilter, result)


**getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly, customFilter, threadedCommentsCollapsed)**

Get a page of filtered and sorted feed elements with comments in a threaded style from a case feed.

**API Version**

44.0

**Available to Guest Users**

44.0

**Requires Chatter**

Yes

**Signature**

public static ConnectApi.FeedElementPage getFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, Integer elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, Boolean showInternalOnly, String customFilter, Boolean threadedCommentsCollapsed)
Parameters

`communityId`  
Type: `String`  
ID for an Experience Cloud site, internal, or `null`.  

`feedType`  
Type: `ConnectApi.FeedType`  
Value must be `ConnectApi.FeedType.Record`.  

`subjectId`  
Type: `String`  
The ID of a case.  

`recentCommentCount`  
Type: `Integer`  
Maximum number of comments to return with each feed item. The default value is 3.  

`elementsPerBundle`  
Type: `Integer`  
Maximum number of feed elements per bundle. The default and maximum value is 10.  

`density`  
Type: `ConnectApi.FeedDensity`  
Specify the amount of content in a feed.  
- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.  
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.  

`pageParam`  
Type: `String`  
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.  

`pageSize`  
Type: `Integer`  
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.  

`sortParam`  
Type: `ConnectApi.FeedSortOrder`  
Values are:  
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.  
- CreatedDateDesc—Sorts by most recent creation date.  
- LastModifiedDateDesc—Sorts by most recent activity.  
- MostViewed—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.  
- Relevance—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.
If you pass in `null`, the default value `CreatedDateDesc` is used.

**showInternalOnly**
Type: `Boolean`
Specifies whether to show only feed items from internal (non-Experience Cloud site) users (true), or not (false). The default value is `false`.

**customFilter**
Type: `String`
Custom filter that applies only to the case feed. See `customFeedFilter` in the Metadata API Developer Guide for supported values.

**threadedCommentsCollapsed**
Type: `Boolean`
Specifies whether to return threaded comments in a collapsed style (true) or not (false). If you pass in null, the default is false.

**Return Value**
Type: `ConnectApi.FeedElementPage`

**Usage**
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly, customFilter, threadedCommentsCollapsed, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

**getFeedElementsFromFilterFeed(communityId, subjectId, keyPrefix)**
Get feed elements from a feed filtered by a key prefix for a user.

**API Version**
31.0

**Requires Chatter**
Yes

**Signature**
```java
public static ConnectApi.FeedElementPage getFeedElementsFromFilterFeed(String communityId, String subjectId, String keyPrefix)
```
Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subjectId
Type: String
ID of the context user or the alias me.

keyPrefix
Type: String
A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

Return Value
Type: ConnectApi.FeedElementPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetFeedElementsFromFilterFeed(communityId, subjectId, keyPrefix, result)


getFeedElementsFromFilterFeed(communityId, subjectId, keyPrefix, pageParam, pageSize, sortParam)
Get a page of sorted feed elements from a feed filtered by a key prefix for a user.

API Version
31.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedElementPage getFeedElementsFromFilterFeed(String communityId, String subjectId, String keyPrefix, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam)
Parameters

communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

subjectId
  Type: String
  ID of the context user or the alias me.

keyPrefix
  Type: String
  A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

pageParam
  Type: String
  Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
  Type: Integer
  Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
  Type: ConnectApi.FeedSortOrder
  Values are:
  • CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
  • CreatedDateDesc—Sorts by most recent creation date.
  • LastModifedDateDesc—Sorts by most recent activity.
  • MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
  • Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
  If you pass in null, the default value CreatedDateDesc is used.

Return Value
  Type: ConnectApi.FeedElementPage

Usage
  To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
  setTestGetFeedElementsFromFilterFeed(communityId, subjectId, keyPrefix, pageParam, pageSize, sortParam, result)
getFeedElementsFromFilterFeed(communityId, subjectId, keyPrefix, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam)

Get a page of sorted feed elements from a feed filtered by a key prefix for a user. Each feed element contains no more than the specified number of comments.

API Version
31.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedElementPage getFeedElementsFromFilterFeed(String communityId, String subjectId, String keyPrefix, Integer recentCommentCount, Integer elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subjectId
Type: String
ID of the context user or the alias me.

keyPrefix
Type: String
A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 00S and Group objects have a prefix of 0F9.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

elementsPerBundle
Type: Integer
Maximum number of feed elements per bundle. The default and maximum value is 10.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.

- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
• FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
• CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
• CreatedDateDesc—Sorts by most recent creation date.
• LastModifiedDateDesc—Sorts by most recent activity.
• MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
• Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
If you pass in null, the default value CreatedDateDesc is used.

Return Value
Type: ConnectApi.FeedElementPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetFeedElementsFromFilterFeed(communityId, subjectId, keyPrefix, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, result)

getFeedElementsFromFilterFeedUpdatedSince(communityId, subjectId, keyPrefix, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince)
Get a page of feed elements from a feed filtered by a key prefix for a user. Include only feed elements that have been updated since the time specified in the updatedSince parameter.

API Version
31.0
Requires Chatter

Yes

Signature

public static ConnectApi.FeedElementPage getFeedElementsFromFilterFeedUpdatedSince(String communityId, String subjectId, String keyPrefix, Integer recentCommentCount, Integer elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, String updatedSince)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subjectId
Type: String
ID of the context user or the alias me.

keyPrefix
Type: String
A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

elementsPerBundle
Type: Integer
Maximum number of feed elements per bundle. The default and maximum value is 10.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.
updatedSince
  Type: String

Opaque token defining the modification timestamp of the feed and the sort order.
The updatedSince parameter doesn't return feed elements that are created in the same second as the call.

Return Value
Type: ConnectApi.FeedElementPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetFeedElementsFromFilterFeedUpdatedSince(communityId, subjectId, keyPrefix, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, result)

getFeedElementsUpdatedSince(communityId, feedType, recentCommentCount, density, pageParam, pageSize, updatedSince)

Get a page of feed elements from the Company, DirectMessageModeration, Home, and Moderation feeds. Include only feed elements that have been updated since the time specified in the updatedSince parameter. Each feed element contains no more than the specified number of comments.

API Version
31.0

Available to Guest Users
31.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedElementPage getFeedElementsUpdatedSince(String communityId, ConnectApi.FeedType feedType, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, String updatedSince)

Parameters
communityId
  Type: String
ID for an Experience Cloud site, internal, or null.

**feedType**
Type: `ConnectApi.FeedType`
Type of feed. Valid values are Company, DirectMessageModeration, Home, and Moderation.

**recentCommentCount**
Type: `Integer`
Maximum number of comments to return with each feed element. The default value is 3.

**density**
Type: `ConnectApi.FeedDensity`
Specify the amount of content in a feed.
- **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**
Type: `String`
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**
Type: `Integer`
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**updatedSince**
Type: `String`
An opaque token containing information about the last modified date of the feed. Do not construct this token. Retrieve this token from the `updatesToken` property of the `ConnectApi.FeedElementPage` response body.
The `updatedSince` parameter doesn’t return feed elements that are created in the same second as the call.

**Return Value**
Type: `ConnectApi.FeedElementPage`

**Usage**
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

**SEE ALSO:**
- `setTestGetFeedElementsUpdatedSince(communityId, feedType, recentCommentCount, density, pageParam, pageSize, updatedSince, result)`

*Apex Developer Guide: Testing ConnectApi Code*
getFeedElementsUpdatedSince(communityId, feedType, recentCommentCount, density, pageParam, pageSize, updatedSince, filter)

Get a page of filtered feed elements from the Home feed. Include only feed elements that have been updated since the time specified in the updatedSince parameter. Each feed element contains no more than the specified number of comments.

API Version
32.0

Available to Guest Users
32.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedElementPage getFeedElementsUpdatedSince(String communityId, ConnectApi.FeedType feedType, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, String updatedSince, ConnectApi.FeedFilter filter)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
The type of feed. The only valid value is Home.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
• AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
• FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.
**pageSizesize**
Type: `Integer`
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**updatedSince**
Type: `String`
An opaque token containing information about the last modified date of the feed. Do not construct this token. Retrieve this token from the `updatesToken` property of the `ConnectApi.FeedElementPage` response body.

The `updatedSince` parameter doesn’t return feed elements that are created in the same second as the call.

**filter**
Type: `ConnectApi.FeedFilter`
 Specifies the feed filters.

- **AllQuestions**—Feed elements that are questions.
- **AuthoredBy**—Feed elements authored by the user profile owner. This value is valid only for the `UserProfile` feed.
- **CommunityScoped**—Feed elements that are scoped to Experience Cloud sites. Currently, these feed elements have a User or a Group parent record. However, other parent record types could be scoped to sites in the future. Feed elements that are always visible in all sites are filtered out. This value is valid only for the `UserProfile` feed.
- **QuestionsWithCandidateAnswers**—Feed elements that are questions that have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **QuestionsWithCandidateAnswersReviewedPublished**—Feed elements that are questions that have candidate answers that have been reviewed or published. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **Read**—Feed elements that are older than 30 days or are marked as read for the context user. Includes existing feed elements when the context user joined the group. This value is valid only for the `Record` feed of a group.
- **SolvedQuestions**—Feed elements that are questions and that have a best answer.
- **UnansweredQuestions**—Feed elements that are questions and that don’t have any answers.
- **UnansweredQuestionsWithCandidateAnswers**—Feed elements that are questions that don’t have answers but have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **Unread**—Feed elements that are created in the past 30 days and aren’t marked as read for the context user. This value is valid only for the `Record` feed of a group.
- **UnsolvedQuestions**—Feed elements that are questions and that don’t have a best answer.

**Return Value**
Type: `ConnectApi.FeedElementPage`
Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

`setTestGetFeedElementsUpdatedSince(communityId, feedType, recentCommentCount, density, pageParam, pageSize, updatedSince, filter, result)`

*Apex Developer Guide: Testing ConnectApi Code*

`getFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince)`

Get a page of feed elements from the Files, Groups, News, People, and Record feeds. Include only feed elements that have been updated since the time specified in the `updatedSince` parameter. Each feed element contains no more than the specified number of comments.

API Version
31.0

Available to Guest Users
31.0

Requires Chatter
Yes

Signature

```java
public static ConnectApi.FeedElementPage getFeedElementsUpdatedSince(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, String updatedSince)
```

Parameters

`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

`feedType`
Type: `ConnectApi.FeedType`
One of these values:
- Files
- Groups
- News
- People
- **Record**

  **subjectId**
  Type: String

  If `feedType` is `ConnectApi.Record`, `subjectId` can be any record ID, including a group ID. Otherwise, it must be the context user or the alias `me`.

- **recentCommentCount**
  Type: Integer

  Maximum number of comments to return with each feed item. The default value is 3.

- **density**
  Type: `ConnectApi.FeedDensity`

  Specify the amount of content in a feed.

  - **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
  - **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

- **pageParam**
  Type: String

  Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

- **pageSize**
  Type: Integer

  Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

- **updatedSince**
  Type: String

  An opaque token containing information about the last modified date of the feed. Do not construct this token. Retrieve this token from the `updatesToken` property of the `ConnectApi.FeedElementPage` response body.

  The `updatedSince` parameter doesn’t return feed elements that are created in the same second as the call.

**Return Value**

Type: `ConnectApi.FeedElementPage`

**Usage**

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

**SEE ALSO:**

`setTestGetFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince, result)`

*Apex Developer Guide: Testing ConnectApi Code*
getFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince, showInternalOnly)

Get a page of feed elements from a record feed. Include only feed elements that have been updated since the time specified in the updatedSince parameter. Specify whether to return feed elements posted by internal (non-Experience Cloud site) users only.

API Version

31.0

Available to Guest Users

31.0

Requires Chatter

Yes

Signature

public static ConnectApi.FeedElementPage getFeedElementsUpdatedSince(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, String updatedSince, Boolean showInternalOnly)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Value must be ConnectApi.FeedType.Record.

subjectId
Type: String
Any record ID, including a group ID.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.

- **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
• FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in **null**, the first page is returned.

**pageSize**
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in **null**, the default size is 25.

**updatedSince**
Type: String
An opaque token containing information about the last modified date of the feed. Do not construct this token. Retrieve this token from the `updatesToken` property of the `ConnectApi.FeedElementPage` response body.

The `updatedSince` parameter doesn’t return feed elements that are created in the same second as the call.

**showInternalOnly**
Type: Boolean
Specifies whether to show only feed elements from internal (non-Experience Cloud site) users (**true**), or not (**false**). The default value is **false**.

**Return Value**
Type: `ConnectApi.FeedElementPage`

**Usage**
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

**SEE ALSO:**
`setTestGetFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince, showInternalOnly, result)`

**Apex Developer Guide: Testing ConnectApi Code**

`getFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, filter)`

Get a page of filtered feed elements from a `UserProfile` feed. Include only feed elements that have been updated since the time specified in the `updatedSince` parameter.

**API Version**
35.0
Available to Guest Users

35.0

Requires Chatter

Yes

Signature

public static ConnectApi.FeedElementPage getFeedElementsUpdatedSince(String communityId,
ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, Integer
elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize,
String updatedSince, ConnectApi.FeedFilter filter)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Value must be ConnectApi.FeedType.UserProfile.

subjectId
Type: String
ID of any user. To specify the context user, use the user ID or the alias me.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

elementsPerBundle
Type: Integer
Maximum number of feed elements per bundle. The default and maximum value is 10.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
• AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
• FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**updatedSince**
Type: `String`
Opaque token defining the modification timestamp of the feed and the sort order.
The `updatedSince` parameter doesn’t return feed elements that are created in the same second as the call.

**filter**
Type: `ConnectApi.FeedFilter`
Value must be `ConnectApi.FeedFilter.CommunityScoped`. Filters the feed to include only feed elements that are scoped to Experience Cloud sites. Feed elements that are always visible in all sites are filtered out.

**Return Value**
Type: `ConnectApi.FeedElementPage`

**Usage**
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

**SEE ALSO:**
- `setTestGetFeedElementsUpdatedSince` (communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, filter, result)

**getFeedElementsUpdatedSince**(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, customFilter)
Get a page of filtered feed elements from a case feed. Include only feed elements that have been updated since the time specified in the `updatedSince` parameter.

**API Version**
40.0

**Available to Guest Users**
40.0

**Requires Chatter**
Yes

**Signature**
```java
public static ConnectApi.FeedElementPage getFeedElementsUpdatedSince(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, Integer elementsPerBundle, Integer density, Integer pageParam, Integer pageSize, Instant updatedSince, ConnectApi.FeedFilter customFilter)
```
elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize,
String updatedSince, String customFilter)

Parameters

communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

feedType
  Type: ConnectApi.FeedType
  Value must be ConnectApi.FeedType.Record.

subjectId
  Type: String
  The ID of a case.

recentCommentCount
  Type: Integer
  Maximum number of comments to return with each feed element. The default value is 3.

elementsPerBundle
  Type: Integer
  Maximum number of feed elements per bundle. The default and maximum value is 10.

density
  Type: ConnectApi.FeedDensity
  Specify the amount of content in a feed.
  • AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
  • FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
  Type: String
  Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
  Type: Integer
  Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

updatedSince
  Type: String
  Opaque token defining the modification timestamp of the feed and the sort order.
  The updatedSince parameter doesn’t return feed elements that are created in the same second as the call.

customFilter
  Type: String
  Custom filter that applies only to the case feed. See customFeedFilter in the Metadata API Developer Guide for supported values.
Return Value
Type: `ConnectApi.FeedElementPage`

Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestGetFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, customFilter, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

`getFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, showInternalOnly)`

Get a page of feed elements from a record feed. Include only feed elements that have been updated since the time specified in the `updatedSince` parameter. Specify the maximum number of feed elements in a bundle and whether to return feed elements posted by internal (non-Experience Cloud site) users only.

API Version
31.0

Available to Guest Users
31.0

Requires Chatter
Yes

Signature
```java
public static ConnectApi.FeedElementPage getFeedElementsUpdatedSince(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, Integer elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, String updatedSince, Boolean showInternalOnly)
```

Parameters
- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.
- `feedType`
  Type: `ConnectApi.FeedType`
  Value must be `ConnectApi.FeedType.Record`.
subjectId
Type: String
Any record ID, including a group ID.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

elementsPerBundle
Type: Integer
Maximum number of feed elements per bundle. The default and maximum value is 10.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

updatedSince
Type: String
An opaque token containing information about the last modified date of the feed. Do not construct this token. Retrieve this token from the updatesToken property of the ConnectApi.FeedElementPage response body.

The updatedSince parameter doesn’t return feed elements that are created in the same second as the call.

showInternalOnly
Type: Boolean
Specifies whether to show only feed elements from internal (non-Experience Cloud site) users (true), or not (false). The default value is false.

Return Value
Type: ConnectApi.FeedElementPage
Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, showInternalOnly, result)

getFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, showInternalOnly, filter)
Get a page of filtered feed elements from a record feed. Include only feed elements that have been updated since the time specified in the updatedSince parameter. Specify the maximum number of feed elements in a bundle and whether to return feed elements posted by internal (non-Experience Cloud site) users only.

API Version
32.0

Available to Guest Users
32.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedElementPage getFeedElementsUpdatedSince(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, Integer elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, String updatedSince, Boolean showInternalOnly, ConnectApi.FeedFilter filter)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Value must be ConnectApi.FeedType.Record.

subjectId
Type: String
Any record ID, including a group ID.
**recentCommentCount**
- **Type:** Integer
- Maximum number of comments to return with each feed element. The default value is 3.

**elementsPerBundle**
- **Type:** Integer
- Maximum number of feed elements per bundle. The default and maximum value is 10.

**density**
- **Type:** ConnectApi.FeedDensity
- Specify the amount of content in a feed.
  - **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
  - **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**
- **Type:** String
- Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**
- **Type:** Integer
- Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**updatedSince**
- **Type:** String
- An opaque token containing information about the last modified date of the feed. Do not construct this token. Retrieve this token from the `updatesToken` property of the `ConnectApi.FeedElementPage` response body.
  - The `updatedSince` parameter doesn’t return feed elements that are created in the same second as the call.

**showInternalOnly**
- **Type:** Boolean
- Specifies whether to show only feed elements from internal (non-Experience Cloud site) users (`true`), or not (`false`). The default value is `false`.

**filter**
- **Type:** ConnectApi.FeedFilter
- Specifies the feed filters.
  - **AllQuestions**—Feed elements that are questions.
  - **AuthoredBy**—Feed elements authored by the user profile owner. This value is valid only for the `UserProfile` feed.
  - **CommunityScoped**—Feed elements that are scoped to Experience Cloud sites. Currently, these feed elements have a User or a Group parent record. However, other parent record types could be scoped to sites in the future. Feed elements that are always visible in all sites are filtered out. This value is valid only for the `UserProfile` feed.
  - **QuestionsWithCandidateAnswers**—Feed elements that are questions that have candidate answers associated with them. This value is valid only for users with the `Access Einstein-Generated Answers` permission.
- **QuestionsWithCandidateAnswersReviewedPublished**—Feed elements that are questions that have candidate answers that have been reviewed or published. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **Read**—Feed elements that are older than 30 days or are marked as read for the context user. Includes existing feed elements when the context user joined the group. This value is valid only for the Record feed of a group.
- **SolvedQuestions**—Feed elements that are questions and that have a best answer.
- **UnansweredQuestions**—Feed elements that are questions and that don't have any answers.
- **UnansweredQuestionsWithCandidateAnswers**—Feed elements that are questions that don’t have answers but have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **Unread**—Feed elements that are created in the past 30 days and aren’t marked as read for the context user. This value is valid only for the Record feed of a group.
- **UnsolvedQuestions**—Feed elements that are questions and that don’t have a best answer.

**Return Value**

Type: `ConnectApi.FeedElementPage`

**Usage**

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

**SEE ALSO:**

`setTestGetFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, showInternalOnly, filter, result)`

*Apex Developer Guide: Testing ConnectApi Code*

**getFeedElementsUpdatedSince**(`communityId`, `feedType`, `subjectId`, `recentCommentCount`, `elementsPerBundle`, `density`, `pageParam`, `pageSize`, `updatedSince`, `showInternalOnly`, `customFilter`)

Get a page of filtered feed elements from a case feed. Include only feed elements that have been updated since the time specified in the `updatedSince` parameter.

**API Version**

40.0

**Available to Guest Users**

40.0

**Requires Chatter**

Yes
signature

```java
public static ConnectApi.FeedElementPage getFeedElementsUpdatedSince(String communityId,
ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, Integer
elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize,
String updatedSince, Boolean showInternalOnly, String customFilter)
```

parameters

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **feedType**
  - Type: `ConnectApi.FeedType`
  - Value must be `ConnectApi.FeedType.Record`.

- **subjectId**
  - Type: `String`
  - The ID of a case.

- **recentCommentCount**
  - Type: `Integer`
  - Maximum number of comments to return with each feed element. The default value is 3.

- **elementsPerBundle**
  - Type: `Integer`
  - Maximum number of feed elements per bundle. The default and maximum value is 10.

- **density**
  - Type: `ConnectApi.FeedDensity`
  - Specify the amount of content in a feed.
    - **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
    - **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

- **pageParam**
  - Type: `String`
  - Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

- **pageSize**
  - Type: `Integer`
  - Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

- **updatedSince**
  - Type: `String`
  - An opaque token containing information about the last modified date of the feed. Do not construct this token. Retrieve this token from the `updatesToken` property of the `ConnectApi.FeedElementPage` response body.
  - The `updatedSince` parameter doesn’t return feed elements that are created in the same second as the call.
showInternalOnly
Type: Boolean
Specifies whether to show only feed elements from internal (non-Experience Cloud site) users (true), or not (false). The default value is false.

customFilter
Type: String
Custom filter that applies only to the case feed. See customFeedFilter in the Metadata API Developer Guide for supported values.

Return Value
Type: ConnectApi.FeedElementPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, showInternalOnly, customFilter, result)

getFeedWithFeedElements(communityId, feedType, pageSize)
Get information about a feed and a page of feed elements from the feed.

API Version
40.0

Available to Guest Users
40.0

Requires Chatter
Yes

Signature
public static ConnectApi.Feed getFeedWithFeedElements(String communityId, ConnectApi.FeedType feedType, Integer pageSize)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.
feedType
Type: `ConnectApi.FeedType`

The type of feed. Valid values are Company, DirectMessageModeration, DirectMessages, Home, Landing, Moderation, and PendingReview. Landing is valid only when `communityId` is internal.

pageSize
Type: `Integer`

Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in 0, feed elements aren’t returned with the feed.

Return Value
Type: `ConnectApi.Feed`

`getFeedWithFeedElements(communityId, feedType, pageSize, recentCommentCount)`

Get a page of information about the feed and the feed elements with the specified number of comments per feed element from the feed.

API Version
40.0

Available to Guest Users
40.0

Requires Chatter
Yes

Signature

```java
public static ConnectApi.Feed getFeedWithFeedElements(String communityId,
ConnectApi.FeedType feedType, Integer pageSize, Integer recentCommentCount)
```

Parameters

`communityId`
Type: `String`

ID for an Experience Cloud site, internal, or `null`.

`feedType`
Type: `ConnectApi.FeedType`

The type of feed. Valid values are Company, DirectMessageModeration, DirectMessages, Home, Landing, Moderation, and PendingReview. Landing is valid only when `communityId` is internal.

`pageSize`
Type: `Integer`
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in 0, feed elements aren’t returned with the feed.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

Return Value
Type: ConnectApi.Feed

getFilterFeed(communityId, subjectId, keyPrefix)
Get a feed filtered by a key prefix for a user.

API Version
28.0

Requires Chatter
Yes

Signature
public static ConnectApi.Feed getFilterFeed(String communityId, String subjectId, String keyPrefix)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subjectId
Type: String
ID of the context user or the alias me.

keyPrefix
Type: String
A key prefix is the first three characters of a record ID, which specifies the object type.

Return Value
Type: ConnectApi.Feed

getFilterFeed(communityId, subjectId, keyPrefix, sortParam)
Get a sorted feed filtered by a key prefix for a user.
API Version
28.0

Requires Chatter
Yes

Signature
public static ConnectApi.Feed getFilterFeed(String communityId, String subjectId, String keyPrefix, ConnectApi.FeedType sortParam)

Parameters

communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

subjectId
  Type: String
  ID of the context user or the alias me.

keyPrefix
  Type: String
  A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

sortParam
  Type: ConnectApi.FeedType
  Values are:
  • CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
  • CreatedDateDesc—Sorts by most recent creation date.
  • LastModifiedDateDesc—Sorts by most recent activity.
  • MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
  • Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
  If you pass in null, the default value CreatedDateDesc is used.

Return Value
Type: ConnectApi.Feed

getFilterFeedDirectory(communityId, subjectId)
Get a feed directory of filter feeds available to the context user.
API Version
30.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedDirectory getFilterFeedDirectory(String communityId, String subjectId)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subjectId
Type: String
ID of the context user or the alias me.

Return Value
Type: ConnectApi.FeedDirectory
This feed directory contains a list of filter feeds, which are the news feed filtered to include feed items whose parent is a specific entity type.

Usage
Call this method to return a directory containing a list of ConnectApi.FeedDirectoryItem objects. Each object contains a key prefix associated with an entity type the context user is following. A key prefix is the first three characters of a record ID, which specifies the object type.

Use key prefixes to filter the news feed so that it contains only feed items whose parent is the entity type associated with the key prefix. For example, get all the feed items whose parent is an Account. To get the feed items, pass a key prefix to the ConnectApi.getFeedItemsFromFilterFeed method.

The information about filter feeds never contains the key prefixes for users (005) or groups (0F9), but all users can use those key prefixes as filters.

The ConnectApi.FeedDirectory.favorites property is always empty when returned by a call to getFilterFeedDirectory because you can’t filter a news feed by favorites.

Example
This example calls getFilterFeedDirectory and loops through the returned FeedDirectoryItem objects to find the key prefixes the context user can use to filter their news feed. It then copies each keyPrefix value to a list. Finally, it passes one of
the key prefixes from the list to the `getFeedItemsFromFilterFeed` method. The returned feed items include every feed item from the news feed whose parent is the entity type specified by the passed key prefix.

```java
String communityId = null;
String subjectId = 'me';

// Create a list to populate with key prefixes.
List<String> keyPrefixList = new List<String>();

// Prepopulate with User and Group record types
// which are available to all users.
keyPrefixList.add('005');
keyPrefixList.add('0F9');

System.debug(keyPrefixList);

// Get the key prefixes available to the context user.
ConnectApi.FeedDirectory myFeedDirectory =
    ConnectApi.ChatterFeeds.getFilterFeedDirectory(null, 'me');

// Loop through the returned feeds list.
for (ConnectApi.FeedDirectoryItem i : myFeedDirectory.feeds) {
    // Grab each key prefix and add it to the list.
    keyPrefixList.add(i.keyPrefix);
}
System.debug(keyPrefixList);

// Use a key prefix from the list to filter the feed items in the news feed.
ConnectApi.FeedItemPage myFeedItemPage =
    ConnectApi.ChatterFeeds.getFeedItemsFromFilterFeed(communityId, subjectId, keyPrefixList[0]);
System.debug(myFeedItemPage);
```

**getLike(communityId, likeId)**

Get a like on a post or comment.

**API Version**

28.0

**Available to Guest Users**

32.0

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.ChatterLike getLike(String communityId, String likeId)
```
Parameters

communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

likeId
  Type: String
  ID for a like.

Return Value

Type: ConnectApi.ChatterLike

getLikesForComment(communityId, commentId)

Get likes for a comment.

API Version

28.0

Available to Guest Users

31.0

Requires Chatter

Yes

Signature

public static ConnectApi.ChatterLikePage getLikesForComment(String communityId, String commentId)

Parameters

communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

commentId
  Type: String
  ID for a comment.

Return Value

Type: ConnectApi.ChatterLikePage
getLikesForComment(communityId, commentId, pageParam, pageSize)
Get a page of likes for a comment.

API Version
28.0

Available to Guest Users
31.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterLikePage getLikesForComment(String communityId, String commentId, Integer pageParam, Integer pageSize)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

commentId
Type: String
ID for a comment.

pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

Return Value
Type: ConnectApi.ChatterLikePage

getLikesForFeedElement(communityId, feedElementId)
Get likes for a feed element.

API Version
32.0
Available to Guest Users

32.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.ChatterLikePage getLikesForFeedElement(String communityId,
                                                      String feedElementId)
```

Parameters

- `communityId`
  
  Type: `String`
  
  ID for an Experience Cloud site, internal, or `null`.

- `feedElementId`
  
  Type: `String`
  
  ID of the feed element.

Return Value

Type: `ConnectApi.ChatterLikePage`

If the feed element doesn’t support the `ChatterLikes` capability, the return value is `ConnectApi.NotFoundException`.

```java
getLikesForFeedElement(communityId, feedElementId, pageParam, pageSize)
```

Get a page of likes for a feed element.

API Version

32.0

Available to Guest Users

32.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.ChatterLikePage getLikesForFeedElement(String communityId,
                                                      String feedElementId, Integer pageParam, Integer pageSize)
```
Parameters

`communityId`
- Type: String
- ID for an Experience Cloud site, internal, or `null`.

`feedElementId`
- Type: String
- ID of the feed element.

`pageParam`
- Type: String
- Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

`pageSize`
- Type: Integer
- Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

Return Value

Type: `ConnectApi.ChatterLikePage`

If the feed element doesn’t support the ChatterLikes capability, the return value is `ConnectApi.NotFoundException`.

`getLinkMetadata(communityId, urls)`

Get link metadata for URLs.

API Version

42.0

Available to Guest Users

42.0

Requires Chatter

No

Signature

`public static ConnectApi.LinkMetadataCollection getLinkMetadata(String communityId, String urls)`

Parameters

`communityId`
- Type: String
- ID for an Experience Cloud site, internal, or `null`.
urls
  Type: String
  Comma-separated list of URL-encoded URLs.

Return Value
Type: ConnectApi.LinkMetadataCollection

getPinnedFeedElementsFromFeed(communityId, feedType, subjectId)
Get pinned feed elements from a group or topic feed.

API Version
41.0

Available to Guest Users
41.0

Requires Chatter
Yes

Signature
public static ConnectApi.PinnedFeedElements getPinnedFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId)

Parameters
communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

feedType
  Type: ConnectApi.FeedType
  The type of feed. Valid values are Record and Topics.

subjectId
  Type: String
  If feedType is Record, subjectId must be a group ID. If feedType is Topics, subjectId must be a topic ID.

Return Value
Type: ConnectApi.PinnedFeedElements
If the feed doesn’t support this capability, the return value is ConnectApi.NotFoundException.
Usage
In the UI, pinned feed elements don’t show all auxiliary information, such as comments, likes, interaction counts, or read by information. As a result, the ConnectApi.PinnedFeedElements output class doesn’t include all the information for these capabilities.

**getReadByForFeedElement(communityId, feedElementId)**
Get information about who read a feed element and when.

**API Version**
40.0

**Requires Chatter**
Yes

**Signature**
```
public static ConnectApi.ReadByPage getReadByForFeedElement(String communityId, String feedElementId)
```

**Parameters**
- **communityId**
  Type: String
  ID for an Experience Cloud site, internal, or **null**.
- **feedElementId**
  Type: String
  ID of the feed element.

**Return Value**
Type: `ConnectApi.ReadByPage`
If the feed element doesn’t support this capability, the return value is `ConnectApi.NotFoundException`.

**getReadByForFeedElement(communityId, feedElementId, pageParam, pageSize)**
Get a page of information about who read a feed element and when.

**API Version**
40.0

**Requires Chatter**
Yes
**Signature**

public static ConnectApi.ReadByPage getReadByForFeedElement(String communityId, String feedElementId, Integer pageParam, Integer pageSize)

**Parameters**

*communityId*
Type: String
ID for an Experience Cloud site, internal, or null.

*feedElementId*
Type: String
ID of the feed element.

*pageParam*
Type: String
Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in null, the first page is returned.

*pageSize*
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

**Return Value**

Type: ConnectApi.ReadByPage
If the feed element doesn’t support this capability, the return value is ConnectApi.NotFoundException.

**getRelatedPosts(communityId, feedElementId, filter, maxResults)**
Get posts related to the context feed element.

**API Version**

37.0

**Available to Guest Users**

37.0

**Requires Chatter**

Yes

**Signature**

public static ConnectApi.RelatedFeedPosts getRelatedPosts(String communityId, String feedElementId, ConnectApi.RelatedFeedPostType filter, Integer maxResults)
Parameters

**communityId**
Type: String
ID for an Experience Cloud site, internal, or null.

**feedElementId**
Type: String
ID of the feed element. The feed element must be a question.

**filter**
Type: `ConnectApi.RelatedFeedPostType`
Specifies the type of related post. Values are:

- **Answered**—Related questions that have at least one answer.
- **BestAnswer**—Related questions that have a best answer.
- **Generic**—All types of related questions, including answered, with a best answer, and unanswered.
- **Unanswered**—Related questions that don’t have answers.

`Generic` is the default value.

**maxResults**
Type: Integer
The maximum number of results to return. You can return up to 25 results; 5 is the default.

Return Value
Type: `ConnectApi.RelatedFeedPosts`
In version 37.0 and later, related feed posts are questions.

Each related feed post has a score indicating how closely it’s related to the context feed post. We return related feed posts sorted by score, with the highest score first.

Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

**getStream(communityId, streamId)**
Get information about a Chatter feed stream.

API Version
39.0

Requires Chatter
Yes
Signature

public static ConnectApi.ChatterStream getStream(String communityId, String streamId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

streamId
Type: String
ID of the Chatter feed stream.

Return Value
Type: ConnectApi.ChatterStream

getStream(communityId, streamId, globalScope)
Get information about a Chatter feed stream, regardless of Experience Cloud site.

API Version
41.0

Requires Chatter
Yes

Signature

public static ConnectApi.ChatterStream getStream(String communityId, String streamId, Boolean globalScope)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

streamId
Type: String
ID of the Chatter feed stream.

globalScope
Type: Boolean
Specifies whether to get streams from all the context user’s Experience Cloud sites, regardless of the communityId value.

Tip: If you know the communityId for the stream, we recommend setting globalScope to false.
Return Value
Type: `ConnectApi.ChatterStream`

`getStreams(communityId)`
Get the Chatter feed streams for the context user.

API Version
39.0

Requires Chatter
Yes

Signature
```
public static ConnectApi.ChatterStreamPage getStreams(String communityId)
```

Parameters
```
communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.
```

Return Value
Type: `ConnectApi.ChatterStreamPage`

`getStreams(communityId, sortParam)`
Get and sort the Chatter feed streams for the context user.

API Version
40.0

Requires Chatter
Yes

Signature
```
public static ConnectApi.ChatterStreamPage getStreams(String communityId, ConnectApi.SortOrder sortParam)
```

Parameters
```
communityId
  Type: String
```


ID for an Experience Cloud site, internal, or null.

**sortParam**
Type: `ConnectApi.SortOrder`
Specifies the sort order. Values are:
- **Ascending**—Items are in ascending alphabetical order (A-Z).
- **Descending**—Items are in descending alphabetical order (Z-A).
- **MostRecentlyViewed**—Items are in descending chronological order by view. This sort order is valid only for Chatter feed streams.
If not specified, default value is **Ascending**.

**Return Value**
Type: `ConnectApi.ChatterStreamPage`

**getStreams(communityId, pageParam, pageSize)**
Get a page of Chatter feed streams for the context user.

**API Version**
39.0

**Requires Chatter**
Yes

**Signature**

```java
public static ConnectApi.ChatterStreamPage getStreams(String communityId, Integer pageParam, Integer pageSize)
```

**Parameters**

- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or null.

- `pageParam`
  Type: `Integer`
  Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

- `pageSize`
  Type: `Integer`
  Specifies the number of items per page. Valid values are from 1 to 250. The default size is 25.

**Return Value**
Type: `ConnectApi.ChatterStreamPage`
getStreams(communityId, pageParam, pageSize, sortParam)
Get a sorted page of Chatter feed streams for the context user.

API Version
40.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterStreamPage getStreams(String communityId, Integer pageParam, Integer pageSize, ConnectApi.SortOrder sortParam)

Parameters
communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

pageParam
  Type: Integer
  Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
  Type: Integer
  Specifies the number of items per page. Valid values are from 1 to 250. The default size is 25.

sortParam
  Type: ConnectApi.SortOrder
  Specifies the sort order. Values are:
  • Ascending—Items are in ascending alphabetical order (A-Z).
  • Descending—Items are in descending alphabetical order (Z-A).
  • MostRecentlyViewed—Items are in descending chronological order by view. This sort order is valid only for Chatter feed streams.

  If not specified, default value is Ascending.

Return Value
Type: ConnectApi.ChatterStreamPage

getStreams(communityId, pageParam, pageSize, sortParam, globalScope)
Get a sorted page of Chatter feed streams from all Enterprise Cloud sites for the context user.
API Version
41.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterStreamPage getStreams(String communityId, Integer pageParam, Integer pageSize, ConnectApi.SortOrder sortParam, Boolean globalScope)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 to 250. The default size is 25.

sortParam
Type: ConnectApi.SortOrder
Specifies the sort order. Values are:
• Ascending—Items are in ascending alphabetical order (A-Z).
• Descending—Items are in descending alphabetical order (Z-A).
• MostRecentlyViewed—Items are in descending chronological order by view. This sort order is valid only for Chatter feed streams.
If not specified, default value is Ascending.

globalScope
Type: Boolean
Specifies whether to get streams from all the context user’s Experience Cloud sites, regardless of the communityId value.

Tip: If you know the communityId for the streams, we recommend setting globalScope to false.

Return Value
Type: ConnectApi.ChatterStreamPage

getSupportedEmojis()
Get supported emojis for the org.
API Version
39.0

Requires Chatter
Yes

Signature
public static ConnectApi.SupportedEmojis getSupportedEmojis()

Return Value
Type: ConnectApi.SupportedEmojis

Usage
To get the list, emojis must be enabled in your org.

getThreadsForFeedComment(communityId, commentId)
Get threaded comments for a comment.

API Version
44.0

Available to Guest Users
44.0

Requires Chatter
Yes

Signature
public static ConnectApi.CommentPage getThreadsForFeedComment(String communityId, String commentId)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

commentId
Type: String
ID of the comment.
Return Value
Type: ConnectApi.CommentPage
If the comment doesn’t support the comments capability, the return value is ConnectApi.NotFoundException.

getThreadsForFeedComment(communityId, commentId, pageParam, pageSize)
Get a page of threaded comments for a comment.

API Version
44.0

Available to Guest Users
44.0

Requires Chatter
Yes

Signature
public static ConnectApi.CommentPage getThreadsForFeedComment(String communityId, String commentId, String pageParam, Integer pageSize)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

commentId
Type: String
ID of the comment.

pageParam
Type: String
Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

Return Value
Type: ConnectApi.CommentPage
If the comment doesn’t support the comments capability, the return value is ConnectApi.NotFoundException.
getThreadsForFeedComment(communityId, commentId, threadedCommentsCollapsed)

Access the comments capability for a comment.

API Version
44.0

Available to Guest Users
44.0

Requires Chatter
Yes

Signature
public static ConnectApi.CommentsCapability getThreadsForFeedComment(String communityId, String commentId, Boolean threadedCommentsCollapsed)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

commentId
Type: String
ID of the comment.

threadedCommentsCollapsed
Type: Boolean
Specifies whether to return threaded comments in a collapsed style (true) or not (false). If you pass in null, the default is false.

Return Value
Type: ConnectApi.CommentsCapability
If the comment doesn't support the comments capability, the return value is ConnectApi.NotFoundException.

getTopUnansweredQuestions(communityId) (Pilot)

Get top unanswered questions for the context user in an Experience Cloud site.

Note: We provided top-five unanswered questions to selected customers through a pilot program that required agreement to specific terms and conditions. This pilot program is closed and not accepting new participants.

API Version
42.0
Requires Chatter
Yes

Signature
public static ConnectApi.FeedElementPage getTopUnansweredQuestions(String communityId)

Parameters

communityId
Type: String
ID of the Experience Cloud site.

Return Value
Type: ConnectApi.FeedElementPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetTopUnansweredQuestions(communityId, result) (Pilot)

getTopUnansweredQuestions(communityId, filter) (Pilot)
Get filtered top unanswered questions for the context user in an Experience Cloud site.

Note: We provided top-five unanswered questions to selected customers through a pilot program that required agreement to specific terms and conditions. This pilot program is closed and not accepting new participants.

API Version
42.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedElementPage getTopUnansweredQuestions(String communityId, ConnectApi.TopUnansweredQuestionsFilterType filter)
Parameters

communityId
Type: String
ID of the Experience Cloud site.

filter
Type: ConnectApi.FeedFilter
Specifies the filter for the feed. UnansweredQuestionsWithCandidateAnswers is the only valid value.

Return Value
Type: ConnectApi.FeedElementPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetTopUnansweredQuestions(communityId, filter, result) (Pilot)

getTopUnansweredQuestions(communityId, pageSize) (Pilot)
Get a page of top unanswered questions for the context user in an Experience Cloud site.

Note: We provided top-five unanswered questions to selected customers through a pilot program that required agreement to specific terms and conditions. This pilot program is closed and not accepting new participants.

API Version
42.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedElementPage getTopUnansweredQuestions(String communityId, Integer pageSize)

Parameters

communityId
Type: String
ID of the Experience Cloud site.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 0 through 10. If you pass in `null`, the default size is 5.

Return Value
Type: `ConnectApi.FeedElementPage`

Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestGetTopUnansweredQuestions(communityId, pageSize, result)` (Pilot)

`getTopUnansweredQuestions(communityId, filter, pageSize)` (Pilot)
Get a page of filtered top unanswered questions for the context user in an Experience Cloud site.

Note: We provided top-five unanswered questions to selected customers through a pilot program that required agreement to specific terms and conditions. This pilot program is closed and not accepting new participants.

API Version
42.0

Requires Chatter
Yes

Signature
```
public static ConnectApi.FeedElementPage getTopUnansweredQuestions(String communityId,
ConnectApi.FeedFilter filter, Integer pageSize)
```

Parameters
- `communityId`
  Type: `String`
  ID of the Experience Cloud site.
- `filter`
  Type: `ConnectApi.FeedFilter`
  Specifies the filter for the feed. `UnansweredQuestionsWithCandidateAnswers` is the only valid value.
- `pageSize`
  Type: `Integer`
  Specifies the number of items per page. Valid values are from 0 through 10. If you pass in `null`, the default size is 5.
Return Value
Type: `ConnectApi.FeedElementPage`

Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestGetTopUnansweredQuestions(communityId, filter, pageSize, result)` (Pilot)
  *Apex Developer Guide: Testing ConnectApi Code*

`getVotesForComment(communityId, commentId, vote)`
Get the first page of users who upvoted or downvoted a comment.

API Version
42.0

Available to Guest Users
42.0

Requires Chatter
Yes

Signature
`public static ConnectApi.VotePage getVotesForComment(String communityId, String commentId, ConnectApiUpDownVoteValue vote)`

Parameters
- `communityId`
  *Type: String*
  ID for an Experience Cloud site, internal, or `null`.
- `commentId`
  *Type: String*
  ID of the comment.
- `vote`
  *Type: ConnectApiUpDownVoteValue*
  Specifies the value of the vote for the feed element. Values are:
  - Down
  - Up
You can’t specify None.

Return Value
Type: ConnectApi.VotePage
If the comment doesn’t support this capability, the return value is ConnectApi.NotFoundException.

getVotesForComment(communityId, commentId, vote, pageParam, pageSize)
Get a page of users who upvoted or downvoted a comment.

API Version
42.0

Available to Guest Users
42.0

Requires Chatter
Yes

Signature
public static ConnectApi.VotePage getVotesForComment(String communityId, String commentId, ConnectApi.UpDownVoteValue vote, Integer pageParam, Integer pageSize)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

commentId
Type: String
ID of the comment.

vote
Type: ConnectApi.UpDownVoteValue
Specifies the value of the vote for the feed element. Values are:
• Down
• Up
You can’t specify None.

pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.
**PageSize**

Type: `Integer`

Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**Return Value**

Type: `ConnectApi.VotePage`

If the comment doesn’t support this capability, the return value is `ConnectApi.NotFoundException`.

**getVotesForFeedElement(communityId, feedElementId, vote)**

Get the first page of users who upvoted or downvoted a feed element.

**API Version**

42.0

**Available to Guest Users**

42.0

**Requires Chatter**

Yes

**Signature**

```
public static ConnectApi.VotePage getVotesForFeedElement(String communityId, String feedElementId, ConnectApi.UpDownVoteValue vote)
```

**Parameters**

`communityId`

Type: `String`

ID for an Experience Cloud site, `internal`, or `null`.

`feedElementId`

Type: `String`

ID of the feed element.

`vote`

Type: `ConnectApi.UpDownVoteValue`

Specifies the value of the vote for the feed element. Values are:

- **Down**
- **Up**

You can’t specify `None`. 
Return Value
Type: ConnectApi.VotePage
If the feed element doesn’t support this capability, the return value is ConnectApi.NotFoundException.

getVotesForFeedElement(communityId, feedElementId, vote, pageParam, pageSize)
Get a page of users who upvoted or downvoted a feed element.

API Version
42.0

Available to Guest Users
42.0

Requires Chatter
Yes

Signature
public static ConnectApi.VotePage getVotesForFeedElement(String communityId, String feedElementId, ConnectApi.UpDownVoteValue vote, Integer pageParam, Integer pageSize)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element.

vote
Type: ConnectApi.UpDownVoteValue
Specifies the value of the vote for the feed element. Values are:
• Down
• Up
You can’t specify None.

pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.
Return Value
Type: `ConnectApi.VotePage`
If the feed element doesn't support this capability, the return value is `ConnectApi.NotFoundException`.

`isCommentEditableByMe(communityId, commentId)`
Discover whether the context user can edit a comment.

API Version
34.0

Requires Chatter
Yes

Signature
```java
public static ConnectApi.FeedEntityIsEditable isCommentEditableByMe(String communityId, String commentId)
```

Parameters
- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.
- **commentId**
  - Type: `String`
  - ID of the comment.

Return Value
Type: `ConnectApi.FeedEntityIsEditable`
If the comment doesn't support the edit capability, the return value is `ConnectApi.NotFoundException`.

SEE ALSO:
- Edit a Comment

`isFeedElementEditableByMe(communityId, feedElementId)`
Discover whether the context user can edit a feed element.

API Version
34.0
Requires Chatter
Yes

Signature

```java
public static ConnectApi.FeedEntityIsEditable isFeedElementEditableByMe(String communityId, String feedElementId)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- `feedElementId`
  - Type: `String`
  - ID of the feed element. Feed items are the only type of feed element that can be edited.

Return Value

- Type: `ConnectApi.FeedEntityIsEditable`
- If the feed element doesn't support the edit capability, the return value is `ConnectApi.NotFoundException`.

SEE ALSO:
- Edit a Feed Element
- Edit a Question Title and Post

`isModified(communityId, feedType, subjectId, since)`

Discover whether a news feed has been updated or changed. Use this method to poll a news feed for updates.

⚠️ **Important:** This feature is available through a Feed Polling pilot program. This pilot program is closed and not accepting new participants.

API Version

- 28.0

Requires Chatter
Yes

Signature

```java
public static ConnectApi.FeedModifiedInfo isModified(String communityId, ConnectApi.FeedType feedType, String subjectId, String since)
```
Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Specifies the type of feed. The only supported type is News

subjectId
Type: String
ID of the context user or the alias me.

since
Type: String
An opaque token containing information about the last modified date of the feed. Retrieve this token from the FeedElementPage.isModifiedToken property.

Return Value
Type: ConnectApi.FeedModifiedInfo

likeComment(communityId, commentId)
Like a comment for the context user.

API Version
28.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterLike likeComment(String communityId, String commentId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

commentId
Type: String
ID for a comment.
Return Value
Type: `ConnectApi.ChatterLike`
If the context user has already liked the comment, this method is a non-operation and returns the existing like.

`likeFeedElement(communityId, feedElementId)`
Like a feed element.

API Version
32.0

Requires Chatter
Yes

Signature
```java
public static ConnectApi.ChatterLike likeFeedElement(String communityId, String feedElementId)
```

Parameters
- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.
- `feedElementId`
  Type: `String`
  ID of the feed element.

Return Value
Type: `ConnectApi.ChatterLike`
If the feed element doesn’t support the ChatterLikes capability, the return value is `ConnectApi.NotFoundException`.

Example
```java
ConnectApi.ChatterLike chatterLike = ConnectApi.ChatterFeeds.likeFeedElement(null, '0D5D0000000KuGh');
```

`postCommentToFeedElement(communityId, feedElementId, text)`
Post a plain-text comment to a feed element.

API Version
32.0
Requires Chatter
Yes

Signature
public static ConnectApi.Comment postCommentToFeedElement(String communityId, String feedElementId, String text)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element.

text
Type: String
Text of the comment. A comment can contain up to 10,000 characters.

Return Value
Type: ConnectApi.Comment
If the feed element doesn’t support the Comments capability, the return value is ConnectApi.NotFoundException.

Example
ConnectApi.Comment comment = ConnectApi.ChatterFeeds.postCommentToFeedElement(null, '0D5D0000000KuGh', 'I agree with the proposal.' );

postCommentToFeedElement(communityId, feedElementId, comment, feedElementFileUpload)
Post a rich-text comment to a feed element. Use this method to include mentions and to attach a file.

API Version
32.0

Requires Chatter
Yes
Signature

```java
public static ConnectApi.Comment postCommentToFeedElement(String communityId, String feedElementId, ConnectApi.CommentInput comment, ConnectApi.BinaryInput feedElementFileUpload)
```

Parameters

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- **feedElementId**
  - Type: String
  - ID of the feed element.

- **comment**
  - Type: ConnectApi.CommentInput
  - The comment body, including text and mentions, and capabilities, such as information about an attached file. A comment can contain up to 10,000 characters.

- **feedElementFileUpload**
  - Type: ConnectApi.BinaryInput
  - A new binary file to attach to the comment, or null. If you specify a binary file, specify the title and description of the file in the comment parameter.

Return Value

- Type: ConnectApi.Comment
  - If the feed element doesn't support the Comments capability, the return value is ConnectApi.NotFoundException.

Example for Posting a Comment with Mentions

You can post comments with mentions two ways. Use the ConnectApiHelper repository on GitHub to write a single line of code, or use this method example.

```java
String communityId = null;
String feedElementId = '0D5D0000000KtW3';

ConnectApi.CommentInput commentInput = new ConnectApi.CommentInput();
ConnectApi.MentionSegmentInput mentionSegmentInput = new ConnectApi.MentionSegmentInput();
ConnectApi.TextSegmentInput textSegmentInput = new ConnectApi.TextSegmentInput();
messageBodyInput.messageSegments = new List<ConnectApi.MessageSegmentInput>();

messageBodyInput.messageSegments.add(textSegmentInput);
textSegmentInput.text = 'Does anyone in this group have an idea? ';
messageBodyInput.messageSegments.add(textSegmentInput);
mentionSegmentInput.id = '005D0000000oOT';
messageBodyInput.messageSegments.add(mentionSegmentInput);
```
commentInput.body = messageBodyInput;

ConnectApi.Comment commentRep = ConnectApi.ChatterFeeds.postCommentToFeedElement(communityId, feedElementId, commentInput, null);

Example for Posting a Comment with an Existing File

String feedElementId = '0D5D0000000KtW3';

ConnectApi.CommentInput commentInput = new ConnectApi.CommentInput();

ConnectApi.TextSegmentInput textSegmentInput = new ConnectApi.TextSegmentInput();

textSegmentInput.text = 'I attached this file from Salesforce Files.';
messageBodyInput.messageSegments = new List<ConnectApi.MessageSegmentInput>();
messageBodyInput.messageSegments.add(textSegmentInput);
commentInput.body = messageBodyInput;

ConnectApi.CommentCapabilitiesInput commentCapabilitiesInput = new ConnectApi.CommentCapabilitiesInput();
ConnectApi.ContentCapabilityInput contentCapabilityInput = new ConnectApi.ContentCapabilityInput();

commentCapabilitiesInput.content = contentCapabilityInput;
contentCapabilityInput.contentDocumentId = '069D0000001rNJ';

commentInput.capabilities = commentCapabilitiesInput;

ConnectApi.Comment commentRep =
ConnectApi.ChatterFeeds.postCommentToFeedElement(Network.getNetworkId(), feedElementId, commentInput, null);

Example for Posting a Comment with a New File

String feedElementId = '0D5D0000000KtW3';

ConnectApi.CommentInput commentInput = new ConnectApi.CommentInput();

ConnectApi.TextSegmentInput textSegmentInput = new ConnectApi.TextSegmentInput();

textSegmentInput.text = 'Enjoy this new file.';
messageBodyInput.messageSegments = new List<ConnectApi.MessageSegmentInput>();
messageBodyInput.messageSegments.add(textSegmentInput);
commentInput.body = messageBodyInput;

ConnectApi.CommentCapabilitiesInput commentCapabilitiesInput = new ConnectApi.CommentCapabilitiesInput();
ConnectApi.ContentCapabilityInput contentCapabilityInput = new ConnectApi.ContentCapabilityInput();

587
Example for Posting a Rich-Text Comment with an Inline Image

You can post rich-text comments with inline images and mentions two ways. Use the ConnectApiHelper repository on GitHub to write a single line of code, or use this method example. In this example, the image file is existing content that has already been uploaded to Salesforce.

String communityId = null;
String feedElementId = '0DSR0000000SBEr';
String imageId = '069R00000000IgQ';
String mentionedUserId = '005R0000000DiMz';

ConnectApi.CommentInput input = new ConnectApi.CommentInput();
ConnectApi.TextSegmentInput textSegment;
ConnectApi.MentionSegmentInput mentionSegment;
ConnectApi.MarkupBeginSegmentInput markupBeginSegment;
ConnectApi.MarkupEndSegmentInput markupEndSegment;
ConnectApi.InlineImageSegmentInput inlineImageSegment;

messageInput.messageSegments = new List<ConnectApi.MessageSegmentInput>();

markupBeginSegment = new ConnectApi.MarkupBeginSegmentInput();
markupBeginSegment.markupType = ConnectApi.MarkupType.Bold;
messageInput.messageSegments.add(markupBeginSegment);

textSegment = new ConnectApi.TextSegmentInput();
textSegment.text = 'Hello ';
messageInput.messageSegments.add(textSegment);

mentionSegment = new ConnectApi.MentionSegmentInput();
mentionSegment.id = mentionedUserId;
messageInput.messageSegments.add(mentionSegment);

textSegment = new ConnectApi.TextSegmentInput();
textSegment.text = '!!';
messageInput.messageSegments.add(textSegment);
Example for Posting a Rich-Text Comment with a Code Block

```java
String communityId = null;
String feedElementId = '0D5R0000000SBEr';
String codeSnippet = '<html>
	<body>
		Hello, world!
	</body>
</html>'

ConnectApi.CommentInput input = new ConnectApi.CommentInput();
ConnectApi.TextSegmentInput textSegment;
ConnectApi.MarkupBeginSegmentInput markupBeginSegment;
ConnectApi.MarkupEndSegmentInput markupEndSegment;

messageInput.messageSegments = new List<ConnectApi.MessageSegmentInput>()

markupBeginSegment = new ConnectApi.MarkupBeginSegmentInput();
markupBeginSegment.markupType = ConnectApi.MarkupType.Code;
messageInput.messageSegments.add(markupBeginSegment);

textSegment = new ConnectApi.TextSegmentInput();
textSegment.text = codeSnippet;
messageInput.messageSegments.add(textSegment);

markupEndSegment = new ConnectApi.MarkupEndSegmentInput();
markupEndSegment.markupType = ConnectApi.MarkupType.Code;
messageInput.messageSegments.add(markupEndSegment);

input.body = messageInput;

ConnectApi.ChatterFeeds.postCommentToFeedElement(communityId, feedElementId, input, null);
```

postFeedElement(communityId, subjectId, feedElementType, text)

Post a plain-text feed element.

API Version

31.0
Requires Chatter
Yes

Signature
public static ConnectApi.FeedElement postFeedElement(String communityId, String subjectId, ConnectApi.FeedElementType feedElementType, String text)

Parameters
communityId
   Type: String
   ID for an Experience Cloud site, internal, or null.

subjectId
   Type: String
   The ID of the parent this feed element is being posted to. This value can be the ID of a user, group, or record, or the string me to indicate the context user.

feedElementType
   Type: ConnectApi.FeedElementType
   The only possible value is FeedItem.

text
   Type: String
   The text of the feed element. A feed element can contain up to 10,000 characters.

Return Value
Type: ConnectApi.FeedElement

Example
ConnectApi.FeedElement feedElement = ConnectApi.ChatterFeeds.postFeedElement(Network.getNetworkId(), '0F9d0000000TreH', ConnectApi.FeedElementType.FeedItem, 'On vacation this week.');

postFeedElement(communityId, feedElement)
Post a rich-text feed element. Include mentions and hashtag topics, attach already uploaded files to a feed element, and associate action link groups with a feed element. You can also use this method to share a feed element and add a comment.

API Version
36.0

Requires Chatter
Yes
Signature

```java
public static ConnectApi.FeedElement postFeedElement(String communityId,
                                                     ConnectApi.FeedElementInput feedElement)
```

Parameters

- `communityId`
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- `feedElement`
  - Type: `ConnectApi.FeedElementInput`
  - Specify rich text, including mentions. Optionally, specify a link, a poll, or up to 10 existing files.

Return Value

- Type: `ConnectApi.FeedElement`

Example for Posting a Feed Element with a Mention

You can post feed elements with mentions two ways. Use the `ConnectApiHelper` repository on GitHub to write a single line of code, or use this method example.

```java
ConnectApi.FeedItemInput feedItemInput = new ConnectApi.FeedItemInput();
ConnectApi.MentionSegmentInput mentionSegmentInput = new ConnectApi.MentionSegmentInput();
ConnectApi.TextSegmentInput textSegmentInput = new ConnectApi.TextSegmentInput();

messageBodyInput.messageSegments = new List<ConnectApi.MessageSegmentInput>();
mentionSegmentInput.id = '005RR000000Dme9';
messageBodyInput.messageSegments.add(mentionSegmentInput);
textSegmentInput.text = 'Could you take a look?';
messageBodyInput.messageSegments.add(textSegmentInput);

feedItemInput.body = messageBodyInput;
feedItemInput.feedElementType = ConnectApi.FeedElementType.FeedItem;
feedItemInput.subjectId = '0F9RR0000004CPw';

ConnectApi.FeedElement feedElement =
ConnectApi.ChatterFeeds.postFeedElement(Network.getNetworkId(), feedItemInput);
```

Example for Posting a Feed Element with Existing Content

```java
// Define the FeedItemInput object to pass to postFeedElement
ConnectApi.FeedItemInput feedItemInput = new ConnectApi.FeedItemInput();
feedItemInput.subjectId = 'me';
ConnectApi.TextSegmentInput textSegmentInput = new ConnectApi.TextSegmentInput();
textSegmentInput.text = 'Would you please review these docs?';
```

---

591
// The MessageBodyInput object holds the text in the post
messageBodyInput.messageSegments = new List<ConnectApi.MessageSegmentInput>();
messageBodyInput.messageSegments.add(textSegmentInput);
feedItemInput.body = messageBodyInput;

// The FeedElementCapabilitiesInput object holds the capabilities of the feed item.
// For this feed item, we define a files capability to hold the file(s).
List<String> fileIds = new List<String>();
fileIds.add('069xx00000000QO');
fileIds.add('069xx00000000QT');
fileIds.add('069xx00000000Qn');
fileIds.add('069xx00000000Qi');
fileIds.add('069xx00000000Qd');

ConnectApi.FilesCapabilityInput filesInput = new ConnectApi.FilesCapabilityInput();
filesInput.items = new List<ConnectApi.FileIdInput>();
for (String fileId : fileIds) {
    ConnectApi.FileIdInput idInput = new ConnectApi.FileIdInput();
    idInput.id = fileId;
    filesInput.items.add(idInput);
}

ConnectApi.FeedElementCapabilitiesInput feedElementCapabilitiesInput = new
ConnectApi.FeedElementCapabilitiesInput();
feedElementCapabilitiesInput.files = filesInput;

feedItemInput.capabilities = feedElementCapabilitiesInput;

// Post the feed item.
ConnectApi.FeedElement feedElement =
ConnectApi.ChatterFeeds.postFeedElement(Network.getNetworkId(), feedItemInput);

Example for Posting a Rich-Text Feed Element with an Inline Image
You can post rich-text feed elements with inline images and mentions two ways. Use the ConnectApiHelper repository on GitHub to write a single line of code, or use this method example. In this example, the image file is existing content that has already been uploaded to Salesforce. The post also includes text and a mention.

String communityId = null;
String imageId = '069D0000000011NA';
String mentionedUserId = '005D0000001Qpr';
String targetUserOrGroupOrRecordId = '005D0000001Gif0';
ConnectApi.FeedItemInput input = new ConnectApi.FeedItemInput();
input.subjectId = targetUserOrGroupOrRecordId;
input.feedElementType = ConnectApi.FeedElementType.FeedItem;
ConnectApi.TextSegmentInput textSegment;
ConnectApi.MentionSegmentInput mentionSegment;
ConnectApi.MarkupBeginSegmentInput markupBeginSegment;
ConnectApi.MarkupEndSegmentInput markupEndSegment;
ConnectApi.InlineImageSegmentInput inlineImageSegment;

messageInput.messageSegments = new List<ConnectApi.MessageSegmentInput>();

markupBeginSegment = new ConnectApi.MarkupBeginSegmentInput();
markupBeginSegment.markupType = ConnectApi.MarkupType.Bold;
messageInput.messageSegments.add(markupBeginSegment);

textSegment = new ConnectApi.TextSegmentInput();
textSegment.text = 'Hello ';
messageInput.messageSegments.add(textSegment);

mentionSegment = new ConnectApi.MentionSegmentInput();
mentionSegment.id = mentionedUserId;
messageInput.messageSegments.add(mentionSegment);

textSegment = new ConnectApi.TextSegmentInput();
textSegment.text = '!';
messageInput.messageSegments.add(textSegment);

markupEndSegment = new ConnectApi.MarkupEndSegmentInput();
markupEndSegment.markupType = ConnectApi.MarkupType.Bold;
messageInput.messageSegments.add(markupEndSegment);

inlineImageSegment = new ConnectApi.InlineImageSegmentInput();
inlineImageSegment.altText = 'image one';
inlineImageSegment.fileId = imageId;
messageInput.messageSegments.add(inlineImageSegment);

input.body = messageInput;
ConnectApi.ChatterFeeds.postFeedElement(communityId, input);

Example for Posting a Rich-Text Feed Element with a Code Block

String communityId = null;
String targetUserOrGroupOrRecordId = 'me';
String codeSnippet = '<html>
	<body>
		Hello, world!
	</body>
</html>';
ConnectApi.FeedItemInput input = new ConnectApi.FeedItemInput();
input.subjectId = targetUserOrGroupOrRecordId;
input.feedElementType = ConnectApi.FeedElementType.FeedItem;

ConnectApi.TextSegmentInput textSegment;
ConnectApi.MarkupBeginSegmentInput markupBeginSegment;
ConnectApi.MarkupEndSegmentInput markupEndSegment;

messageInput.messageSegments = new List<ConnectApi.MessageSegmentInput>();

markupBeginSegment = new ConnectApi.MarkupBeginSegmentInput();
markupBeginSegment.markupType = ConnectApi.MarkupType.Code;
messageInput.messageSegments.add(markupBeginSegment);
textSegment.text = codeSnippet;
messageInput.messageSegments.add(textSegment);

markupEndSegment = new ConnectApiMarkupEndSegmentInput();
markupEndSegment.markupType = ConnectApiMarkupType.Code;
messageInput.messageSegments.add(markupEndSegment);

input.body = messageInput;
ConnectApi.ChatterFeeds.postFeedElement(communityId, input);

Example for Sharing a Feed Element (in Version 39.0 and Later)

// Define the FeedItemInput object to pass to postFeedElement
ConnectApiFeedItemInput feedItemInput = new ConnectApiFeedItemInput();
feedItemInput.subjectId = 'me';
ConnectApiTextSegmentInput textSegmentInput = new ConnectApiTextSegmentInput();
textSegmentInput.text = 'Look at this post I'm sharing.';
// The MessageBodyInput object holds the text in the post
ConnectApiMessageBodyInput messageBodyInput = new ConnectApiMessageBodyInput();
messageBodyInput.messageSegments = new List<ConnectApiMessageSegmentInput>(){
messageBodyInput.messageSegments.add(textSegmentInput);

feedItemInput.body = messageBodyInput;
ConnectApiFeedEntityShareCapabilityInput shareInput = new
ConnectApiFeedEntityShareCapabilityInput();
shareInput.feedEntityId = '0D5R0000000SEbc';
ConnectApiFeedElementCapabilitiesInput feedElementCapabilitiesInput = new
ConnectApiFeedElementCapabilitiesInput();
feedElementCapabilitiesInput.feedEntityShare = shareInput;
feedItemInput.capabilities = feedElementCapabilitiesInput;
// Post the feed item.
ConnectApiFeedElement feedElement =
ConnectApi.ChatterFeeds.postFeedElement(Network.getNetworkId(), feedItemInput);

Example for Sending a Direct Message

// Define the FeedItemInput object to pass to postFeedElement
ConnectApiFeedItemInput feedItemInput = new ConnectApiFeedItemInput();

ConnectApiTextSegmentInput textSegmentInput = new ConnectApiTextSegmentInput();
textSegmentInput.text = 'Thanks for attending my presentation test run this morning. Send me any feedback.';
// The MessageBodyInput object holds the text in the post
ConnectApiMessageBodyInput messageBodyInput = new ConnectApiMessageBodyInput();
messageBodyInput.messageSegments = new List<ConnectApiMessageSegmentInput>(){
messageBodyInput.messageSegments.add(textSegmentInput);

feedItemInput.body = messageBodyInput;
// The FeedElementCapabilitiesInput object holds the capabilities of the feed item.
// For this feed item, we define a direct message capability to hold the member(s) and the subject.
List<String> memberIds = new List<String>();
memberIds.add('005B00000016OUQ');
memberIds.add('005B0000001rIN6');

ConnectApi.DirectMessageCapabilityInput dmInput = new
ConnectApi.DirectMessageCapabilityInput();
dmInput.subject = 'Thank you!';
dmInput.membersToAdd = memberIds;

ConnectApi.FeedElementCapabilitiesInput feedElementCapabilitiesInput = new
ConnectApi.FeedElementCapabilitiesInput();
feedElementCapabilitiesInput.directMessage = dmInput;

feedItemInput.capabilities = feedElementCapabilitiesInput;

// Post the feed item.
ConnectApi.FeedElement feedElement =
ConnectApi.ChatterFeeds.postFeedElement(Network.getNetworkId(), feedItemInput);

postFeedElementBatch(communityId, feedElements)
Post a list of feed elements.

API Version
32.0

Requires Chatter
Yes

Signature
public static ConnectApi.BatchResult[] postFeedElementBatch(String communityId,
List<ConnectApi.BatchInput> feedElements)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.
feedElements
Type: List<ConnectApi.BatchInput>
The list can contain up to 500 ConnectApi.BatchInput objects. In the ConnectApi.BatchInput constructor,
the input object must be a concrete instance of the abstract ConnectApi.FeedElementInput class.

Return Value
Type: ConnectApi.BatchResult[]}
The `ConnectApi.BatchResult.getResult()` method returns a `ConnectApi.FeedElement` object. The returned objects correspond to each of the input objects and are returned in the same order as the input objects. The method call fails only if an error occurs that affects the entire operation (such as a parsing failure). If an individual object causes an error, the error is embedded within the `ConnectApi.BatchResult` list.

Usage

Use this method to post a list of feed elements efficiently. Create a list containing up to 500 objects and insert them all for the cost of one DML statement.

In version 36.0 and later, you can attach only one already uploaded file to each post. The `ConnectApi.BatchInput` has three constructors, but the `postFeedElementBatch` method supports only `ConnectApi.BatchInput(Object input)` in version 35.0 and later. This constructor doesn't support a binary input.

In version 32.0–35.0, this method supports both `ConnectApi.BatchInput(Object input)` and `ConnectApi.BatchInput(Object input, ConnectApi.BinaryInput binary)` constructors. The `ConnectApi.BatchInput(Object input, ConnectApi.BinaryInput binary)` constructor allows for a single binary input.

In each constructor, the input object must be an instance of `ConnectApi.FeedElementInput`.

Example

This trigger bulk posts to the feeds of newly inserted accounts.

```apex
trigger postFeedItemToAccount on Account (after insert) {
    Account[] accounts = Trigger.new;

    // Bulk post to the account feeds.
    List<ConnectApi.Batch> batchInputs = new List<ConnectApi.Batch>();

    for (Account a : accounts) {
        ConnectApi.FeedItemInput input = new ConnectApi.FeedItemInput();

        input.subjectId = a.id;

        body.messageSegments = new List<ConnectApi.MessageSegmentInput>();
        ConnectApi.TextSegmentInput textSegment = new ConnectApi.TextSegmentInput();
        textSegment.text = 'Let\'s win the ' + a.name + ' account.';

        body.messageSegments.add(textSegment);
        input.body = body;

        ConnectApi.BatchInput batchInput = new ConnectApi.BatchInput(input);
        batchInputs.add(batchInput);
    }

    ConnectApi.ChatterFeeds.postFeedElementBatch(Network.getNetworkId(), batchInputs);
}
```
**publishDraftFeedElement(communityId, feedElementId, feedElement)**

Publish a draft feed element.

**API Version**

44.0

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.FeedElement publishDraftFeedElement(String communityId, String feedElementId, ConnectApi.FeedElementInput feedElement)
```

**Parameters**

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.
- **feedElementId**
  - Type: `String`
  - ID of the feed element to publish.
- **feedElement**
  - Type: `ConnectApi.FeedElementInput`
  - Use this optional parameter to edit your draft before publishing.

**Return Value**

- Type: `ConnectApi.FeedElement`
  - The published feed element has a new ID.

**searchFeedElements(communityId, q)**

Get the first page of feed elements that match the search criteria.

**API Version**

31.0

**Available to Guest Users**

31.0

**Requires Chatter**

Yes
Signature

```java
public static ConnectApi.FeedElementPage searchFeedElements(String communityId, String q)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- `q`
  - Type: `String`
  - Required and can't be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

Return Value

Type: `ConnectApi.FeedElementPage`

Usage

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- `setTestSearchFeedElements(communityId, q, result)`

`searchFeedElements(communityId, q, sortParam)`

Get the first page of sorted feed elements that match the search criteria.

API Version

31.0

Available to Guest Users

31.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.FeedElementPage searchFeedElements(String communityId, String q, ConnectApi.FeedSortOrder sortParam)
```
Parameters

communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

q
  Type: String
  Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

sortParam
  Type: ConnectApi.FeedSortOrder
  Values are:
  • CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
  • CreatedDateDesc—Sorts by most recent creation date.
  • LastModifiedDateDesc—Sorts by most recent activity.
  • MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
  • Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
  If you pass in null, the default value CreatedDateDesc is used.

Return Value
  Type: ConnectApi.FeedElementPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
  setTestSearchFeedElements(communityId, q, sortParam, result)

searchFeedElements(communityId, q, threadedCommentsCollapsed)
Get the feed elements and comments that match the search criteria.

API Version
  44.0

Available to Guest Users
  44.0
Requires Chatter
Yes

Signature

```java
public static ConnectApi.FeedElementPage searchFeedElements(String communityId, String q, Boolean threadedCommentsCollapsed)
```

Parameters

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- **q**
  - Type: String
  - Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

- **threadedCommentsCollapsed**
  - Type: Boolean
  - Specifies whether to return threaded comments in a collapsed style (true) or not (false). If you pass in null, the default is false.

Return Value

Type: `ConnectApi.FeedElementPage`

Usage

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- `setTestSearchFeedElements(communityId, q, threadedCommentsCollapsed, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

```java
searchFeedElements(communityId, q, pageParam, pageSize)
```

Get a page of feed elements that match the search criteria.

API Version

31.0

Available to Guest Users

31.0
**Requires Chatter**
Yes

**Signature**

```
public static ConnectApi.FeedElementPage searchFeedElements(String communityId, String q, String pageParam, Integer pageSize)
```

**Parameters**

- **communityId**
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.

- **q**
  Type: `String`
  Required and can’t be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See **Wildcards**.

- **pageParam**
  Type: `String`
  Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

- **pageSize**
  Type: `Integer`
  Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**Return Value**

Type: `ConnectApi.FeedElementPage`

**Usage**

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

**SEE ALSO:**
- `setTestSearchFeedElements(communityId, q, pageParam, pageSize, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

**searchFeedElements(communityId, q, pageParam, pageSize, sortParam)**

Get a page of sorted feed elements that match the search criteria.

**API Version**

31.0
Available to Guest Users
31.0

Requires Chatter
Yes

Signature

public static ConnectApi.FeedElementPage searchFeedElements(String communityId, String q, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
• CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
• CreatedDateDesc—Sorts by most recent creation date.
• LastModifiedDateDesc—Sorts by most recent activity.
• MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
• Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

If you pass in null, the default value CreatedDateDesc is used.

Return Value
Type: ConnectApi.FeedElementPage
Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestSearchFeedElements(communityId, q, pageParam, pageSize, sortParam, result)

searchFeedElements(communityId, q, pageParam, pageSize, threadedCommentsCollapsed)
Get a page of feed elements with comments in a threaded style that match the search criteria.

API Version
44.0

Available to Guest Users
44.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedElementPage searchFeedElements(String communityId, String q, String pageParam, Integer pageSize, Boolean threadedCommentsCollapsed)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.
threadedCommentsCollapsed
    Type: Boolean
    Specifies whether to return threaded comments in a collapsed style (true) or not (false). If you pass in null, the default is false.

Return Value
Type: ConnectApi.FeedElementPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
    setTestSearchFeedElements(communityId, q, pageParam, pageSize, threadedCommentsCollapsed, result)


searchFeedElements(communityId, q, recentCommentCount, pageParam, pageSize, sortParam)
Get a page of sorted feed elements that match the search criteria. Each feed element includes no more than the specified number of comments.

API Version
31.0

Available to Guest Users
31.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedElementPage searchFeedElements(String communityId, String q, Integer recentCommentCount, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam)

Parameters
communityId
    Type: String
    ID for an Experience Cloud site, internal, or null.
q
  Type: String
  Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

recentCommentCount
  Type: Integer
  Maximum number of comments to return with each feed element. The default value is 3.

pageParam
  Type: String
  Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
  Type: Integer
  Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
  Type: ConnectApi.FeedSortOrder
  Values are:
  • CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
  • CreatedDateDesc—Sorts by most recent creation date.
  • LastModifiedDateDesc—Sorts by most recent activity.
  • MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
  • Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
  If you pass in null, the default value CreatedDateDesc is used.

Return Value
  Type: ConnectApi.FeedElementPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
  setTestSearchFeedElements(communityId, q, recentCommentCount, pageParam, pageSize, sortParam, result)

searchFeedElementsInFeed(communityId, feedType, q)
Get the feed elements from the Company, DirectMessageModeration, Home, Moderation, and PendingReview feeds that match the search criteria.
API Version
31.0

Available to Guest Users
31.0

Requires Chatter
Yes

Signature

public static ConnectApi.FeedElementPage searchFeedElementsInFeed(String communityId, ConnectApi.FeedType feedType, String q)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Type of feed. Valid values are Company, DirectMessageModeration, Home, Moderation, and PendingReview.

q
Type: String
Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

Return Value
Type: ConnectApi.FeedElementPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

setTestSearchFeedElementsInFeed(communityId, feedType, q, result)

searchFeedElementsInFeed(communityId, feedType, pageParam, pageSize, sortParam, q)
Get a page of sorted feed elements from the Company, DirectMessageModeration, Home, Moderation, and PendingReview feeds that match the search criteria.
API Version
31.0

Available to Guest Users
31.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedElementPage searchFeedElementsInFeed(String communityId, ConnectApi.FeedType feedType, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Type of feed. Valid values are Company, DirectMessageModeration, Home, Moderation, and PendingReview.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

If you pass in null, the default value CreatedDateDesc is used.
q
Type: String
Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

Return Value
Type: ConnectApi.FeedElementPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestSearchFeedElementsInFeed(communityId, feedType, pageParam, pageSize, sortParam, q, result)

searchFeedElementsInFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, q)
Get a page of sorted feed elements from the Company, DirectMessageModeration, Home, Moderation, and PendingReview feeds that match the search criteria. Each feed element includes no more than the specified number of comments.

API Version
31.0

Available to Guest Users
31.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedElementPage searchFeedElementsInFeed(String communityId, ConnectApi.FeedType feedType, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.
**FeedType**
Type: `ConnectApi.FeedType`
Type of feed. Valid values are Company, DirectMessageModeration, Home, Moderation, and PendingReview.

**recentCommentCount**
Type: `Integer`
Maximum number of comments to return with each feed element. The default value is 3.

**density**
Type: `ConnectApi.FeedDensity`
Specify the amount of content in a feed.
- **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**
Type: `String`
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**
Type: `Integer`
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**
Type: `ConnectApi.FeedSortOrder`
Values are:
- **CreateDateAsc**—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- **CreateDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for Home feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- **Relevance**—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
If you pass in `null`, the default value `CreateDateDesc` is used.

**q**
Type: `String`
Required and can't be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

**Return Value**
Type: `ConnectApi.FeedElementPage`
Usage

To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

setTestSearchFeedElementsInFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, q, result)


searchFeedElementsInFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, q, filter)

Get a page of sorted and filtered feed elements from the Home feed that match the search criteria. Each feed element includes no more than the specified number of comments.

API Version

32.0

Available to Guest Users

32.0

Requires Chatter

Yes

Signature

public static ConnectApi.FeedElementPage searchFeedElementsInFeed(String communityId, ConnectApi.FeedType feedType, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q, ConnectApi.FeedFilter filter)

Parameters

communityId

Type: String

ID for an Experience Cloud site, internal, or null.

feedType

Type: ConnectApi.FeedType

The type of feed. The only valid value is Home.

recentCommentCount

Type: Integer

Maximum number of comments to return with each feed element. The default value is 3.
density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.
When the sortParam is MostViewed, you must pass in null for the pageParam.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.
When the sortParam is MostViewed, the pageSize must be a value from 1 to 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
If you pass in null, the default value CreatedDateDesc is used.

q
Type: String
Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

filter
Type: ConnectApi.FeedFilter
Specifies the feed filters.
- AllQuestions—Feed elements that are questions.
- AuthoredBy—Feed elements authored by the user profile owner. This value is valid only for the UserProfile feed.
- CommunityScoped—Feed elements that are scoped to Experience Cloud sites. Currently, these feed elements have a User or a Group parent record. However, other parent record types could be scoped to sites in the future. Feed elements that are always visible in all sites are filtered out. This value is valid only for the UserProfile feed.
- QuestionsWithCandidateAnswers—Feed elements that are questions that have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
• QuestionsWithCandidateAnswersReviewedPublished—Feed elements that are questions that have candidate answers that have been reviewed or published. This value is valid only for users with the Access Einstein-Generated Answers permission.
• Read—Feed elements that are older than 30 days or are marked as read for the context user. Includes existing feed elements when the context user joined the group. This value is valid only for the Record feed of a group.
• SolvedQuestions—Feed elements that are questions and that have a best answer.
• UnansweredQuestions—Feed elements that are questions and that don’t have any answers.
• UnansweredQuestionsWithCandidateAnswers—Feed elements that are questions that don’t have answers but have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
• Unread—Feed elements that are created in the past 30 days and aren’t marked as read for the context user. This value is valid only for the Record feed of a group.
• UnsolvedQuestions—Feed elements that are questions and that don’t have a best answer.

Return Value
Type: ConnectApi.FeedElementPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestSearchFeedElementsInFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, q, filter, result)

searchFeedElementsInFeed(communityId, feedType, subjectId, q)
Search up to 5,000 of the most recent feed elements in a feed for a subject ID that match the search string. Feed elements are returned in order of most recent activity.

API Version
31.0

Available to Guest Users
31.0

Requires Chatter
Yes
Signature

```java
public static ConnectApi.FeedElementPage searchFeedElementsInFeed(String communityId,
ConnectApi.FeedType feedType, String subjectId, String q)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- `feedType`
  - Type: `ConnectApi.FeedType`
  - Type of feed. Valid values include every `ConnectApi.FeedType` except `Company`, `DirectMessages`, `Filter`, `Landing`, `Streams`, and `Topics`.

- `subjectId`
  - Type: `String`
  - If `feedType` is `Record`, `subjectId` can be any record ID, including a group ID. If `feedType` is `UserProfile`, `subjectId` can be any user ID. If the `feedType` is any other value, `subjectId` must be the ID of the context user or the alias `me`.

- `q`
  - Type: `String`
  - Required and can't be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

Return Value

Type: `ConnectApi.FeedElementPage`

Usage

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- `setTestSearchFeedElementsInFeed(communityId, feedType, subjectId, q, result)`

```java
searchFeedElementsInFeed(communityId, feedType, subjectId, pageParam,
pageSize, sortParam, q)
```

Get a page of sorted feed elements from a feed for a record or user that match the search criteria.

API Version

31.0
Available to Guest Users

31.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.FeedElementPage searchFeedElementsInFeed(String communityId,
ConnectApi.FeedType feedType, String subjectId, String pageParam, Integer pageSize,
ConnectApi.FeedSortOrder sortParam, String q)
```

Parameters

`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

`feedType`
Type: `ConnectApi.FeedType`
Type of feed. Valid values include every `ConnectApi.FeedType` except `Company`, `DirectMessages`, `Filter`, `Landing`, `Streams`, and `Topics`.

`subjectId`
Type: `String`
If `feedType` is `Record`, `subjectId` can be any record ID, including a group ID. If `feedType` is `UserProfile`, `subjectId` can be any user ID. If the `feedType` is any other value, `subjectId` must be the ID of the context user or the alias me.

`pageParam`
Type: `String`
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

`pageSize`
Type: `Integer`
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

`sortParam`
Type: `ConnectApi.FeedSortOrder`
Order of feed items in the feed.
- `CreatedDateAsc`—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- `CreatedDateDesc`—Sorts by most recent creation date.
- `LastModifiedDateDesc`—Sorts by most recent activity.
- `MostViewed`—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- `Relevance`—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.
If you pass in null, the default value CreatedDateDesc is used.

q
Type: String

Search term. Searches keywords in the user or group name. A minimum of one character is required. This parameter doesn’t support wildcards. This parameter is required.

Return Value
Type: ConnectApi.FeedElementPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestSearchFeedElementsInFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam, q, result)


searchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q)
Get a page of sorted feed elements from a feed that match the search criteria. Each feed element includes no more than the specified number of comments.

API Version
31.0

Available to Guest Users
31.0

Requires Chatter
Yes

Signature
public static ConnectApi.FeedElementPage searchFeedElementsInFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

**feedType**
Type: `ConnectApi.FeedType`
Type of feed. Valid values include every `ConnectApi.FeedType` except `Company`, `DirectMessages`, `Filter`, `Landing`, `Streams`, and `Topics`.

**subjectId**
Type: `String`
If `feedType` is `Record`, `subjectId` can be any record ID, including a group ID. If `feedType` is `UserProfile`, `subjectId` can be any user ID. If the `feedType` is any other value, `subjectId` must be the ID of the context user or the alias me.

**recentCommentCount**
Type: `Integer`
Maximum number of comments to return with each feed element. The default value is 3.

**density**
Type: `ConnectApi.FeedDensity`
Specify the amount of content in a feed.
- **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**
Type: `String`
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**
Type: `Integer`
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**
Type: `ConnectApi.FeedSortOrder`
Values are:
- **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- **Relevance**—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.

If you pass in `null`, the default value `CreatedDateDesc` is used.

**q**
Type: `String`
Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

Return Value
Type: `ConnectApi.FeedElementPage`

Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestSearchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, result)`

searchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, filter)
Get a page of sorted and filtered feed elements from a UserProfile feed that match the search criteria.

API Version
35.0

Available to Guest Users
35.0

Requires Chatter
Yes

Signature
```java
public static ConnectApi.FeedElementPage searchFeedElementsInFeed(String communityId,
                      ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount,
                      ConnectApi.FeedDensity density, String pageParam, Integer pageSize,
                      ConnectApi.FeedSortOrder sortParam, String q, ConnectApi.FeedFilter filter)
```

Parameters
- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or null.
- `feedType`
  Type: `ConnectApi.FeedType`
Value must be `ConnectApi.FeedType.UserProfile`.

**subjectId**
Type: `String`
ID of any user. To specify the context user, use the user ID or the alias `me`.

**recentCommentCount**
Type: `Integer`
Maximum number of comments to return with each feed element. The default value is 3.

**density**
Type: `ConnectApi.FeedDensity`
Specify the amount of content in a feed.
- **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**
Type: `String`
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**
Type: `Integer`
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**
Type: `ConnectApi.FeedSortOrder`
Values are:
- **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- **Relevance**—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.

If you pass in `null`, the default value `CreatedDateDesc` is used.

**q**
Type: `String`
One or more keywords to search for in the feed elements visible to the context user. The search string can contain wildcards and must contain at least two characters that aren’t wildcards. See `Wildcards`.

**filter**
Type: `ConnectApi.FeedFilter`
Value must be `ConnectApi.FeedFilter.CommunityScoped`. Filters the feed to include only feed elements that are scoped to Experience Cloud sites. Feed elements that are always visible in all sites are filtered out.
Return Value
Type: `ConnectApi.FeedElementPage`

Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestSearchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, filter, result)`

`searchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, customFilter)`
Get a page of sorted and filtered feed elements from a case feed that match the search criteria.

API Version
40.0

Available to Guest Users
40.0

Requires Chatter
Yes

Signature
```
public static ConnectApi.FeedElementPage searchFeedElementsInFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q, String customFilter)
```

Parameters

- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or null.
- `feedType`
  Type: `ConnectApi.FeedType`
  Value must be `ConnectApi.FeedType.Record`.
- `subjectId`
  Type: `String`
The ID of a case.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
- **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
- **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- **Relevance**—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
If you pass in null, the default value CreatedDateDesc is used.

q
Type: String
One or more keywords to search for in the feed elements visible to the context user. The search string can contain wildcards and must contain at least two characters that aren’t wildcards. See Wildcards.

customFilter
Type: String
Custom filter that applies only to the case feed. See customFeedFilter in the Metadata API Developer Guide for supported values.

Return Value
Type: ConnectApi.FeedElementPage
Usage

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

```
setTestSearchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, customFilter, result)
```

*Apex Developer Guide: Testing ConnectApi Code*

`searchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, showInternalOnly)`

Get a page of sorted feed elements from a feed for a record or user that match the search criteria. Each feed element includes no more than the specified number of comments. Specify whether to return feed elements posted by internal (non-Experience Cloud site) users only.

API Version

31.0

Available to Guest Users

31.0

Requires Chatter

Yes

Signature

```
public static ConnectApi.FeedElementPage searchFeedElementsInFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q, Boolean showInternalOnly)
```

Parameters

`communityId`

Type: `String`

ID for an Experience Cloud site, `internal`, or `null`.

`feedType`

Type: `ConnectApi.FeedType`

Value must be `ConnectApi.FeedType.Record`.

`subjectId`

Type: `String`

Any record ID, including a group ID.
recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
If you pass in null, the default value CreatedDateDesc is used.

q
Type: String
Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

showInternalOnly
Type: Boolean
Specifies whether to show only feed elements from internal (non-Experience Cloud site) users (true), or not (false). The default value is false.

Return Value
Type: ConnectApi.FeedElementPage
Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestSearchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, showInternalOnly, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

`searchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, showInternalOnly, filter)`
Get a page of sorted and filtered feed elements from a feed for a record or user that match the search criteria. Each feed element includes no more than the specified number of comments. Specify whether to return feed elements posted by internal (non-Experience Cloud site) users only.

**API Version**
32.0

**Available to Guest Users**
32.0

**Requires Chatter**
Yes

**Signature**
```java
public static ConnectApi.FeedElementPage searchFeedElementsInFeed(String communityId,
    ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount,
    ConnectApi.FeedDensity density, String pageParam, Integer pageSize,
    ConnectApi.FeedSortOrder sortParam, String q, Boolean showInternalOnly,
    ConnectApi.FeedFilter filter)
```

**Parameters**
- `communityId`
  Type: `String`
  ID for an Experience Cloud site, *internal*, or `null`.
- `feedType`
  Type: `ConnectApi.FeedType`
  Value must be `ConnectApi.FeedType.Record`.
- `subjectId`
  Type: `String`
  Any record ID, including a group ID.
recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
  • AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
  • FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
  • CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
  • CreatedDateDesc—Sorts by most recent creation date.
  • LastModifiedDateDesc—Sorts by most recent activity.
  • MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
  • Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
If you pass in null, the default value CreatedDateDesc is used.

q
Type: String
Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

showInternalOnly
Type: Boolean
Specifies whether to show only feed elements from internal (non-Experience Cloud site) users (true), or not (false). The default value is false.

filter
Type: ConnectApi.FeedFilter
Specifies the feed filters.
  • AllQuestions—Feed elements that are questions.
  • AuthoredBy—Feed elements authored by the user profile owner. This value is valid only for the UserProfile feed.
• CommunityScoped—Feed elements that are scoped to Experience Cloud sites. Currently, these feed elements have a User or a Group parent record. However, other parent record types could be scoped to sites in the future. Feed elements that are always visible in all sites are filtered out. This value is valid only for the UserProfile feed.

• QuestionsWithCandidateAnswers—Feed elements that are questions that have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.

• QuestionsWithCandidateAnswersReviewedPublished—Feed elements that are questions that have candidate answers that have been reviewed or published. This value is valid only for users with the Access Einstein-Generated Answers permission.

• Read—Feed elements that are older than 30 days or are marked as read for the context user. Includes existing feed elements when the context user joined the group. This value is valid only for the Record feed of a group.

• SolvedQuestions—Feed elements that are questions and that have a best answer.

• UnansweredQuestions—Feed elements that are questions and that don’t have any answers.

• UnansweredQuestionsWithCandidateAnswers—Feed elements that are questions that don’t have answers but have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.

• Unread—Feed elements that are created in the past 30 days and aren’t marked as read for the context user. This value is valid only for the Record feed of a group.

• UnsolvedQuestions—Feed elements that are questions and that don’t have a best answer.

**Return Value**

Type: `ConnectApi.FeedElementPage`

**Usage**

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

`setTestSearchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, showInternalOnly, filter, result)`

*Apex Developer Guide: Testing ConnectApi Code*

`searchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, showInternalOnly, customFilter)`

Get a page of sorted and filtered feed elements from a case feed that match the search criteria.

**API Version**

40.0

**Available to Guest Users**

40.0
Requires Chatter
Yes

Signature

```java
public static ConnectApi.FeedElementPage searchFeedElementsInFeed(String communityId,
    ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount,
    ConnectApi.FeedDensity density, String pageParam, Integer pageSize,
    ConnectApi.FeedSortOrder sortParam, String q, Boolean showInternalOnly, String
customFilter)
```

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>communityId</td>
<td>String</td>
<td>ID for an Experience Cloud site, internal, or null.</td>
</tr>
<tr>
<td>feedType</td>
<td>ConnectApi.FeedType</td>
<td>Value must be ConnectApi.FeedType.Record.</td>
</tr>
<tr>
<td>subjectId</td>
<td>String</td>
<td>The ID of a case.</td>
</tr>
<tr>
<td>recentCommentCount</td>
<td>Integer</td>
<td>Maximum number of comments to return with each feed element. The default value is 3.</td>
</tr>
<tr>
<td>density</td>
<td>ConnectApi.FeedDensity</td>
<td>Specify the amount of content in a feed.</td>
</tr>
<tr>
<td>pageParam</td>
<td>String</td>
<td>Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.</td>
</tr>
<tr>
<td>pageSize</td>
<td>Integer</td>
<td>Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.</td>
</tr>
<tr>
<td>sortParam</td>
<td>ConnectApi.FeedSortOrder</td>
<td>Values are:</td>
</tr>
</tbody>
</table>
• CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.

• CreatedDateDesc—Sorts by most recent creation date.

• LastModifiedDateDesc—Sorts by most recent activity.

• MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.

• Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

If you pass in null, the default value CreatedDateDesc is used.

q
Type: String
Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

showInternalOnly
Type: Boolean
Specifies whether to show only feed elements from internal (non-Experience Cloud site) users (true), or not (false). The default value is false.

filter
Type: String
Custom filter that applies only to the case feed. See customFeedFilter in the Metadata API Developer Guide for supported values.

Return Value
Type: ConnectApi.FeedElementPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestSearchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, showInternalOnly, customFilter, result)

searchFeedElementsInFilterFeed(communityId, subjectId, keyPrefix, q)
Get the feed elements from a feed filtered by a key prefix that match the search criteria.

API Version
31.0

Requires Chatter
Yes
Signature

```java
public static ConnectApi.FeedElementPage searchFeedElementsInFilterFeed(String communityId, String subjectId, String keyPrefix, String q)
```

Parameters

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **subjectId**
  - Type: `String`
  - ID of the context user or the alias `me`.

- **keyPrefix**
  - Type: `String`
  - A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

- **q**
  - Type: `String`
  - Required and can’t be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See **Wildcards**.

Return Value

- Type: `ConnectApi.FeedElementPage`

Usage

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- `setTestSearchFeedElementsInFilterFeed(communityId, subjectId, keyPrefix, q, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

**searchFeedElementsInFilterFeed(communityId, subjectId, keyPrefix, pageParam, pageSize, sortParam, q)**

Get a page of sorted feed elements from a feed filtered by a key prefix that match the search criteria.

API Version

- 31.0

Requires Chatter

- Yes
function searchFeedElementsInFilterFeed(communityId: String, subjectId: String, keyPrefix: String, pageParam: String, pageSize: Integer, sortParam: ConnectApi.FeedSortOrder, q: String): ConnectApi.FeedElementPage {
    // Function implementation goes here
}

**Parameters**

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- **subjectId**
  - Type: String
  - ID of the context user or the alias me.

- **keyPrefix**
  - Type: String
  - A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 00S and Group objects have a prefix of 0F9.

- **pageParam**
  - Type: String
  - Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

- **pageSize**
  - Type: Integer
  - Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

- **sortParam**
  - Type: ConnectApi.FeedSortOrder
  - Values are:
    - CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
    - CreatedDateDesc—Sorts by most recent creation date.
    - LastModifiedDateDesc—Sorts by most recent activity.
    - MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
    - Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
  - If you pass in null, the default value CreatedDateDesc is used.

- **q**
  - Type: String
  - Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

**Return Value**

Type: ConnectApi.FeedElementPage
Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
   - `setTestSearchFeedElementsInFilterFeed(communityId, subjectId, keyPrefix, pageParam, pageSize, sortParam, q, result)`
   - *Apex Developer Guide: Testing ConnectApi Code*

`searchFeedElementsInFilterFeed(communityId, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, sortParam, q)`
Get a page of sorted feed elements from a feed filtered by a key prefix that match the search criteria. Each feed element includes no more than the specified number of comments.

API Version
31.0

Requires Chatter
Yes

Signature
```java
public static ConnectApi.FeedElementPage searchFeedElementsInFilterFeed(String communityId, String subjectId, String keyPrefix, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q)
```

Parameters
- `communityId`<br>  Type: `String`<br>  ID for an Experience Cloud site, internal, or `null`.
- `subjectId`<br>  Type: `String`<br>  ID of the context user or the alias `me`.
- `keyPrefix`<br>  Type: `String`<br>  A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 00S and Group objects have a prefix of 0F9.
- `recentCommentCount`<br>  Type: `Integer`<br>  Maximum number of comments to return with each feed element. The default value is 3.
- `density`<br>  Type: `ConnectApi.FeedDensity`
Specify the amount of content in a feed.

- **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**

Type: **String**

Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**

Type: **Integer**

Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**

Type: **ConnectApi.FeedSortOrder**

Values are:

- **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- **Relevance**—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.

If you pass in `null`, the default value `CreatedDateDesc` is used.

**q**

Type: **String**

Required and can’t be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See **Wildcards**.

**Return Value**

Type: **ConnectApi.FeedElementPage**

**Usage**

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

**SEE ALSO:**

- `setTestSearchFeedElementsInFilterFeed(communityId, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, sortParam, q, result)`
- *Apex Developer Guide: Testing ConnectApi Code*
searchStreams(communityId, q)
Search the Chatter feed streams for the context user.

API Version
40.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterStreamPage searchStreams(String communityId, String q)

Parameters
communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

q
  Type: String
  Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

Return Value
Type: ConnectApi.ChatterStreamPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestSearchStreams(communityId, q, result)

searchStreams(communityId, q, sortParam)
Search and sort the Chatter feed streams for the context user.

API Version
40.0
Requires Chatter
Yes

Signature
public static ConnectApi.ChatterStreamPage searchStreams(String communityId, String q, ConnectApi.SortOrder sortParam)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

sortParam
Type: ConnectApi.SortOrder
Specifies the sort order. Values are:
- Ascending—Items are in ascending alphabetical order (A-Z).
- Descending—Items are in descending alphabetical order (Z-A).
- MostRecentlyViewed—Items are in descending chronological order by view. This sort order is valid only for Chatter feed streams.
If not specified, default value is Ascending.

Return Value
Type: ConnectApi.ChatterStreamPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestSearchStreams(communityId, q, sortParam, result)

searchStreams(communityId, q, pageParam, pageSize)
Search the Chatter feed streams for the context user and return a page of results.

API Version
40.0
Requires Chatter
Yes

Signature

```java
public static ConnectApi.ChatterStreamPage searchStreams(String communityId, String q, Integer pageParam, Integer pageSize)
```

Parameters

- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.

- `q`
  Type: `String`
  Required and can't be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See `Wildcards`.

- `pageParam`
  Type: `Integer`
  Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

- `pageSize`
  Type: `Integer`
  Specifies the number of items per page. Valid values are from 1 to 250. The default size is 25.

Return Value

Type: `ConnectApi.ChatterStreamPage`

Usage

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- `setTestSearchStreams(communityId, q, pageParam, pageSize, result)`

`searchStreams(communityId, q, pageParam, pageSize, sortParam)`

Search the Chatter feed streams for the context user and return a sorted page of results.

API Version

40.0
Requires Chatter
Yes

Signature

public static ConnectApi.ChatterStreamPage searchStreams(String communityId, String q, Integer pageParam, Integer pageSize, ConnectApi.SortOrder sortParam)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 to 250. The default size is 25.

sortParam
Type: ConnectApi.SortOrder
Specifies the sort order. Values are:
- Ascending—Items are in ascending alphabetical order (A-Z).
- Descending—Items are in descending alphabetical order (Z-A).
- MostRecentlyViewed—Items are in descending chronological order by view. This sort order is valid only for Chatter feed streams.

If not specified, default value is Ascending.

Return Value
Type: ConnectApi.ChatterStreamPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestSearchStreams(communityId, q, pageParam, pageSize, sortParam, result)
searchStreams(communityId, q, pageParam, pageSize, sortParam, globalScope)

Search the Chatter feed streams from all Experience Cloud sites for the context user and return a sorted page of results.

API Version
41.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterStreamPage searchStreams(String communityId, String q, Integer pageParam, Integer pageSize, ConnectApi.SortOrder sortParam, Boolean globalScope)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 to 250. The default size is 25.

sortParam
Type: ConnectApi.SortOrder
Specifies the sort order. Values are:
• Ascending—Items are in ascending alphabetical order (A-Z).
• Descending—Items are in descending alphabetical order (Z-A).
• MostRecentlyViewed—Items are in descending chronological order by view. This sort order is valid only for Chatter feed streams.

If not specified, default value is Ascending.

globalScope
Type: Boolean
 Specifies whether to get streams from all the context user's Experience Cloud sites, regardless of the communityId value.

Tip: If you know the communityId for the streams, we recommend setting globalScope to false.
Return Value
Type: `ConnectApi.ChatterStreamPage`

Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

```java
setCommentIsVerified(communityId, commentId, isVerified)
```
Mark a comment as verified or unverified.

API Version
41.0

Requires Chatter
Yes

Signature
```java
public static ConnectApi.VerifiedCapability setCommentIsVerified(String communityId, String commentId, Boolean isVerified)
```

Parameters
- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.
- `commentId`
  Type: `String`
  ID of the comment on a question post. Only one comment on a question post can be marked as verified.
- `isVerified`
  Type: `Boolean`
  Specifies whether to mark the comment as verified (`true`) or unverified (`false`). Only verified comments can be marked as unverified, and only unverified comments can be marked as verified.

Return Value
Type: `ConnectApi.VerifiedCapability`
If the comment doesn’t support this capability, the return value is `ConnectApi.NotFoundException`.

```java
setCommentIsVerifiedByAnonymized(communityId, commentId, isVerified, isVerifiedByAnonymized)
```
Mark a comment as verified by an anonymous user.
API Version
43.0

Requires Chatter
Yes

Signature
public static ConnectApi.VerifiedCapability setCommentIsVerifiedByAnonymized(String communityId, String commentId, Boolean isVerified, Boolean isVerifiedByAnonymized)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

commentId
Type: String
ID of the comment on a question post. Only one comment on a question post can be marked as verified.

isVerified
Type: Boolean
Specifies whether to mark the comment as verified (true) or unverified (false).

Only verified comments can be marked as unverified, and only unverified comments can be marked as verified.

isVerifiedByAnonymized
Type: Boolean
Specifies whether to mark the comment as verified by an anonymous user (true).

If a user previously verified a comment and then requested the activity to be deleted, use isVerifiedByAnonymized to maintain the verification and anonymize the value of lastVerifiedByUser.

You can't set isVerified and isVerifiedByAnonymized to true at the same time. isVerifiedByAnonymized can be set to true only if isVerified is already set to true.

You can't set isVerifiedByAnonymized to false. After isVerifiedByAnonymized is set to true, it can be undone only when another user marks the comment as unverified and then reverifies the comment.

Return Value
Type: ConnectApi.VerifiedCapability
If the comment doesn't support this capability, the return value is ConnectApi.NotFoundException.

setCommentVote(communityId, commentId, upDownVote)
Upvote or downvote a comment.
API Version
41.0

Requires Chatter
Yes

Signature
public static ConnectApi.UpDownVoteCapability setCommentVote(String communityId, String commentId, ConnectApi.UpDownVoteCapabilityInput upDownVote)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

commentId
Type: String
ID of the comment.

upDownVote
Type: ConnectApi.UpDownVoteCapabilityInput
A ConnectApi.UpDownVoteCapabilityInput object that includes your vote.

Return Value
Type: ConnectApi.UpDownVoteCapability
If the comment doesn’t support this capability, the return value is ConnectApi.NotFoundException.

setFeedCommentStatus(communityId, commentId, status)
Set the status of a comment.

API Version
38.0

Requires Chatter
Yes

Signature
public static ConnectApi.StatusCapability setFeedCommentStatus(String communityId, String commentId, ConnectApi.StatusCapabilityInput status)
Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

commentId
Type: String
ID of the comment.

status
Type: ConnectApi.StatusCapabilityInput
A ConnectApi.StatusCapabilityInput object that includes the status you want to set.

Return Value
Type: ConnectApi.StatusCapability
If the comment doesn’t support this capability, the return value is ConnectApiNotFoundException.

Usage
Only users with the Can Approve Feed Post and Comment permission can set the status of a feed post or comment.

setFeedElementIsClosed(communityId, feedElementId, isClosed)
Set a feed element to closed.

Users can’t edit (specifically the feed item body or title), comment on, or delete a closed feed element. If the closed feed element is a poll, users can’t vote on it. Users can’t edit (specifically the comment body) or delete a comment on a closed feed element or select or remove it as best answer.

Admins and moderators can edit and delete closed feed elements and comments on closed feed elements. Admins and moderators can select or remove the best answer status on comments on closed feed elements.

API Version
43.0

Requires Chatter
Yes

Signature
public static ConnectApi.CloseCapability setFeedElementIsClosed(String communityId, String feedElementId, Boolean isClosed)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.
feedElementId
  Type: String
  ID of the feed element.

isClosed
  Type: Boolean
  Specifies whether to set the feed element to closed (true) or not (false).

Return Value
Type: ConnectApi.CloseCapability
If the feed element doesn’t support this capability, the return value is ConnectApi.NotFoundException.

setFeedElementVote(communityId, feedElementId, upDownVote)
Upvote or downvote a feed element.

API Version
41.0

Requires Chatter
Yes

Signature
public static ConnectApi.UpDownVoteCapability setFeedElementVote(String communityId, String feedElementId, ConnectApi.UpDownVoteCapabilityInput upDownVote)

Parameters
communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

feedElementId
  Type: String
  ID of the feed element.

upDownVote
  Type: ConnectApi.UpDownVoteCapabilityInput
  A ConnectApi.UpDownVoteCapabilityInput object that includes your vote.

Return Value
Type: ConnectApi.UpDownVoteCapability
If the feed element doesn’t support this capability, the return value is ConnectApi.NotFoundException.
**setFeedEntityStatus(communityId, feedElementId, status)**

Set the status of a feed post.

**API Version**

37.0

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.StatusCapability setFeedEntityStatus(String communityId, String feedElementId, ConnectApi.StatusCapabilityInput status)
```

**Parameters**

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- **feedElementId**
  - Type: String
  - ID of the feed element.

- **status**
  - Type: `ConnectApi.StatusCapabilityInput`
  - A `ConnectApi.StatusCapabilityInput` object that includes the status you want to set.

**Return Value**

Type: `ConnectApi.StatusCapability`

If the feed element doesn’t support this capability, the return value is `ConnectApi.NotFoundException`.

**Usage**

Only users with the Can Approve Feed Post and Comment permission can set the status of a feed post or comment.

**setIsMutedByMe(communityId, feedElementId, isMutedByMe)**

Mute or unmute a feed element.

**API Version**

35.0

**Requires Chatter**

Yes
Signature

public static ConnectApi.MuteCapability setIsMutedByMe(String communityId, String feedElementId, Boolean isMutedByMe)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element.

isMutedByMe
Type: Boolean
Indicates whether the feed element is muted for the context user. Default value is false.

Return Value

Type: ConnectApi.MuteCapability
If the feed element doesn’t support this capability, the return value is ConnectApi.NotFoundException.

setIsReadByMe(communityId, feedElementId, readBy)
Mark a feed element as read for the context user using an input class.

API Version

40.0

Requires Chatter

Yes

Signature

public static ConnectApi.ReadByCapability setIsReadByMe(String communityId, String feedElementId, ConnectApi.ReadByCapabilityInput readBy)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element to mark as read.
**readBy**
Type: `ConnectApi.ReadByCapabilityInput`  
A `ConnectApi.ReadByCapabilityInput` body indicating to mark the feed elements as read.

**Return Value**
Type: `ConnectApi.ReadByCapability`
If the feed element doesn’t support this capability, the return value is `ConnectApi.NotFoundException`.

**setIsReadByMe(communityId, feedElementId, isReadByMe)**  
Mark a feed element as read for the context user.

**API Version**  
40.0

**Requires Chatter**  
Yes

**Signature**

```java
public static ConnectApi.ReadByCapability setIsReadByMe(String communityId, String feedElementId, Boolean isReadByMe)
```

**Parameters**

- **communityId**
  Type: `String`  
  ID for an Experience Cloud site, internal, or `null`.

- **feedElementId**
  Type: `String`  
  ID of the feed element to mark as read.

- **isReadByMe**
  Type: `Boolean`  
  Specifies to mark the feed element as read (`true`) for the context user.

**Return Value**
Type: `ConnectApi.ReadByCapability`
If the feed element doesn’t support this capability, the return value is `ConnectApi.NotFoundException`.

**updateComment(communityId, commentId, comment)**  
Edit a comment.
API Version
34.0

Requires Chatter
Yes

Signature

```java
public static ConnectApi.Comment updateComment(String communityId, String commentId, ConnectApi.CommentInput comment)
```

Parameters

- **communityId**
  Type: String
  ID for an Experience Cloud site, internal, or null.

- **commentId**
  Type: String
  ID of the comment to be edited.

- **comment**
  Type: ConnectApi.CommentInput
  Information about the comment to be edited.

Return Value

Type: ConnectApi.Comment

If the comment doesn’t support the edit capability, the return value is `ConnectApi.NotFoundException`.

Example

```java
String commentId;
String communityId = Network.getNetworkId();

// Get the last feed item created by the context user.
List<FeedItem> feedItems = [SELECT Id FROM FeedItem WHERE CreatedById = :UserInfo.getUserId() ORDER BY CreatedDate DESC];
if (feedItems.isEmpty()) {
    // Return null within anonymous apex.
    return null;
}
String feedElementId = feedItems[0].id;

ConnectApi.CommentPage commentPage =
ConnectApi.ChatterFeeds.getCommentsForFeedElement(communityId, feedElementId);
if (commentPage.items.isEmpty()) {
    // Return null within anonymous apex.
    return null;
}
```
commentId = commentPage.items[0].id;

ConnectApi.FeedEntityIsEditable isEditable =
ConnectApi.ChatterFeeds.isCommentEditableByMe(communityId, commentId);

if (isEditable.isEditableByMe == true){
    ConnectApi.CommentInput commentInput = new ConnectApi.CommentInput();
    ConnectApi.TextSegmentInput textSegmentInput = new ConnectApi.TextSegmentInput();

    messageBodyInput.messageSegments = new List<ConnectApi.MessageSegmentInput>();

    textSegmentInput.text = 'This is my edited comment.';
    messageBodyInput.messageSegments.add(textSegmentInput);

    commentInput.body = messageBodyInput;

    ConnectApi.Comment editedComment = ConnectApi.ChatterFeeds.updateComment(communityId,
    commentId, commentInput);
}

updateDirectMessage(communityId, feedElementId, directMessage)

Update the members of a direct message.

API Version
40.0

Requires Chatter
Yes

Signature
public static ConnectApi.DirectMessageCapability updateDirectMessage(String communityId,
String feedElementId, ConnectApi.DirectMessageCapabilityInput directMessage)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element.

directMessage
Type: ConnectApi.DirectMessageCapabilityInput
A ConnectApi.DirectMessageCapabilityInput body that includes the members to add and remove.
Return Value
Type: `ConnectApi.DirectMessageCapability`
If the feed element doesn’t support this capability, the return value is `ConnectApi.NotFoundException`.

`updateFeedElement(communityId, feedElementId, feedElement)`
Edit a feed element.

API Version
34.0

Requires Chatter
Yes

Signature
```java
public static ConnectApi.FeedElement updateFeedElement(String communityId, String feedElementId, ConnectApi.FeedElementInput feedElement)
```

Parameters
- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.
- `feedElementId`
  Type: `String`
  ID of the feed element to be edited. Feed items are the only type of feed element that can be edited.
- `feedElement`
  Type: `ConnectApi.FeedElementInput`
  Information about the feed item to be edited.

Return Value
Type: `ConnectApi.FeedElement`
If the feed element doesn’t support the edit capability, the return value is `ConnectApi.NotFoundException`.

Example for Editing a Feed Post
```java
String communityId = Network.getNetworkId();

// Get the last feed item created by the context user.
List<FeedItem> feedItems = [SELECT Id FROM FeedItem WHERE CreatedById = :UserInfo.getUserId() ORDER BY CreatedDate DESC];
if (feedItems.isEmpty()) {
    // Return null within anonymous apex.
    return null;
```
String feedElementId = feedItems[0].id;

ConnectApi.FeedEntityIsEditable isEditable =
ConnectApi.ChatterFeeds.isFeedElementEditableByMe(communityId, feedElementId);

if (isEditable.isEditableByMe == true){
    ConnectApi.FeedItemInput feedItemInput = new ConnectApi.FeedItemInput();
    ConnectApi.TextSegmentInput textSegmentInput = new ConnectApi.TextSegmentInput();

    messageBodyInput.messageSegments = new List<ConnectApi.MessageSegmentInput>();

    textSegmentInput.text = 'This is my edited post.';
    messageBodyInput.messageSegments.add(textSegmentInput);

    feedItemInput.body = messageBodyInput;
    ConnectApi.FeedElement editedFeedElement =
    ConnectApi.ChatterFeeds.updateFeedElement(communityId, feedElementId, feedItemInput);
}

Example for Editing a Question Title and Post

String communityId = Network.getNetworkId();

// Get the last feed item created by the context user.
List<FeedItem> feedItems = [SELECT Id FROM FeedItem WHERE CreatedById = :UserInfo.getUserId()
    ORDER BY CreatedDate DESC];
if (feedItems.isEmpty()) {
    // Return null within anonymous apex.
    return null;
}
String feedElementId = feedItems[0].id;

ConnectApi.FeedEntityIsEditable isEditable =
ConnectApi.ChatterFeeds.isFeedElementEditableByMe(communityId, feedElementId);

if (isEditable.isEditableByMe == true){
    ConnectApi.FeedItemInput feedItemInput = new ConnectApi.FeedItemInput();
    ConnectApi.FeedElementCapabilitiesInput feedElementCapabilitiesInput = new
    ConnectApi.FeedElementCapabilitiesInput();
    ConnectApi.QuestionAndAnswersCapabilityInput questionAndAnswersCapabilityInput = new
    ConnectApi.QuestionAndAnswersCapabilityInput();
    ConnectApi.TextSegmentInput textSegmentInput = new ConnectApi.TextSegmentInput();

    messageBodyInput.messageSegments = new List<ConnectApi.MessageSegmentInput>();

    textSegmentInput.text = 'This is my edited question.';
    messageBodyInput.messageSegments.add(textSegmentInput);

    feedItemInput.body = messageBodyInput;
updateFeedElementBookmarks communityId, feedElementId, bookmarks

Bookmark a feed element or remove a bookmark from a feed element using an input class.

API Version
32.0

Requires Chatter
Yes

Signature
public static ConnectApi.BookmarksCapability updateFeedElementBookmarks(String communityId, String feedElementId, ConnectApi.BookmarksCapabilityInput bookmarks)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element.

bookmarks
Type: ConnectApi.BookmarksCapabilityInput
Information about a bookmark.

Return Value
Type: ConnectApi.BookmarksCapability
If the feed element doesn’t support this capability, the return value is ConnectApi.NotFoundException.

updateFeedElementBookmarks communityId, feedElementId, isBookmarkedByCurrentUser

Bookmark a feed element or remove a bookmark from a feed element.
API Version
32.0

Requires Chatter
Yes

Signature
public static ConnectApi.BookmarksCapability updateFeedElementBookmarks(String communityId, String feedElementId, Boolean isBookmarkedByCurrentUser)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element.

isBookmarkedByCurrentUser
Type: Boolean
Specify whether to bookmark the feed element (true) or not (false).

Return Value
Type: ConnectApi.BookmarksCapability
If the feed element doesn’t support this capability, the return value is ConnectApi.NotFoundException.

Example
ConnectApi.BookmarksCapability bookmark =
ConnectApi.ChatterFeeds.updateFeedElementBookmarks(null, '0D5D0000000KuGh', true);

updateFeedElementReadByCapabilityBatch(communityId, feedElementIds, readBy)
Mark multiple feed elements as read by the context user at the same time using an input class.

API Version
40.0

Requires Chatter
Yes
public static ConnectApi.BatchResult[] updateFeedElementReadByCapabilityBatch(String communityId, List<String> feedElementIds, ConnectApi.ReadByCapabilityInput readBy)

Parameters

communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

feedElementIds
  Type: List<String>
  Up to 500 feed element IDs to mark as read.

readBy
  Type: ConnectApi.ReadByCapabilityInput
  A ConnectApi.ReadByCapabilityInput body indicating to mark the feed elements as read.

Return Value

Type: ConnectApi.BatchResult[]
If the feed element doesn't support this capability, the return value is ConnectApiNotFoundException.
The returned objects correspond to each of the input objects and are returned in the same order as the input objects.
The method call fails only if an error occurs that affects the entire operation (such as a parsing failure). If an individual object causes an error, the error is embedded within the ConnectApi.BatchResult list.

updateFeedElementReadByCapabilityBatch(communityId, feedElementIds, isReadByMe)
Mark multiple feed elements as read by the context user at the same time.

API Version
40.0

Requires Chatter
Yes

Signature

public static ConnectApi.BatchResult[] updateFeedElementReadByCapabilityBatch(String communityId, List<String> feedElementIds, Boolean isReadByMe)

Parameters

communityId
  Type: String
ID for an Experience Cloud site, internal, or null.

`feedElementIds`
Type: `List<String>`
Up to 500 feed element IDs to mark as read.

`isReadByMe`
Type: `Boolean`
Specifies to mark the feed element as read (`true`) for the context user.

Return Value
Type: `ConnectApi.BatchResult[]`
If the feed element doesn’t support this capability, the return value is `ConnectApi.NotFoundException`.

**updateLikeForComment(communityId, commentId, isLikedByCurrentUser)**
Like or unlike a comment.

API Version
39.0

Requires Chatter
Yes

Signature
```
public static ConnectApi.ChatterLikePage updateLikeForComment(String communityId, String commentId, Boolean isLikedByCurrentUser)
```

Parameters

`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or null.

`commentId`
Type: `String`
ID of the comment.

`isLikedByCurrentUser`
Type: `Boolean`
Specifies if the context user likes (`true`) or unlikes (`false`) the comment.

Return Value
Type: `ConnectApi.ChatterLikePage`
updateLikeForFeedElement(communityId, feedElementId, isLikedByCurrentUser)
Like or unlike a feed element.

API Version
39.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterLikePage updateLikeForFeedElement(String communityId, String feedElementId, Boolean isLikedByCurrentUser)

Parameters
communityId
 Type: String
 ID for an Experience Cloud site, internal, or null.

feedElementId
 Type: String
 ID of the feed element.

isLikedByCurrentUser
 Type: Boolean
 Specifies if the context user likes (true) or unlikes (false) the feed element.

Return Value
Type: ConnectApi.ChatterLikePage
If the feed element doesn’t support the ChatterLikes capability, the return value is ConnectApi.NotFoundException.

updatePinnedFeedElements(communityId, feedType, subjectId, pin)
Pin or unpin feed elements to a group or topic feed.

API Version
41.0

Available to Guest Users
41.0

Requires Chatter
Yes
Signature

```java
public static ConnectApi.PinCapability updatePinnedFeedElements(String communityId,
        ConnectApi.FeedType feedType, String subjectId, ConnectApi.PinCapabilityInput pin)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or null.

- `feedType`
  - Type: `ConnectApi.FeedType`
  - The type of feed. Valid values are Record and Topics.

- `subjectId`
  - Type: `String`
  - If `feedType` is Record, `subjectId` must be a group ID. If `feedType` is Topics, `subjectId` must be a topic ID.

- `pin`
  - Type: `ConnectApi.PinCapabilityInput`
  - A `ConnectApi.PinCapabilityInput` object indicating the feed element to pin or unpin.

Return Value

- Type: `ConnectApi.PinCapability`

If the feed doesn’t support this capability, the return value is `ConnectApi.NotFoundException`.

```java
updateStream(communityId, streamId, streamInput)
```

Update a Chatter feed stream.

API Version

39.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.ChatterStream updateStream(String communityId, String streamId,
        ConnectApi.ChatterStreamInput streamInput)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or null.
**streamId**
Type: `String`
ID of the Chatter feed stream.

**streamInput**
Type: `ConnectApi.ChatterStreamInput`

**Return Value**
Type: `ConnectApi.ChatterStream`

**voteOnFeedElementPoll**
`communityId`, `feedElementId`, `myChoiceId`
Vote on a poll or change your vote on a poll.

**API Version**
32.0

**Requires Chatter**
Yes

**Signature**
```java
public static ConnectApi.PollCapability voteOnFeedElementPoll(String communityId, String feedElementId, String myChoiceId)
```

**Parameters**

`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

`feedElementId`
Type: `String`
ID of the feed element.

`myChoiceId`
Type: `String`
ID of the poll item you’re voting for. The key prefix for poll items is 09A.

**Return Value**
Type: `ConnectApi.PollCapability`
If the feed element doesn’t support this capability, the return value is `ConnectApiNotFoundException`.
Example

```java
ConnectApi.PollCapability poll = ConnectApi.ChatterFeeds.voteOnFeedElementPoll(null, '0D5D0000000XZaUKAW', '09AD000000000TKMAY');
```

ChatterFeeds Test Methods

The following are the test methods for ChatterFeeds. All methods are static.

For information about using these methods to test your ConnectApi code, see Testing ConnectApi Code.

IN THIS SECTION:

- `setTestGetFeedElementsFromFeed(communityId, feedType, result)`
  - Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElementsFromFeed` is called with matching parameters in a test context. Use the `getFeedElementsFromFeed` method with the same parameters or the code throws an exception.

- `setTestGetFeedElementsFromFeed(communityId, feedType, pageParam, pageSize, sortParam, result)`
  - Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElementsFromFeed` is called with matching parameters in a test context. Use the `getFeedElementsFromFeed` method with the same parameters or the code throws an exception.

- `setTestGetFeedElementsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, result)`
  - Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElementsFromFeed` is called with matching parameters in a test context. Use the `getFeedElementsFromFeed` method with the same parameters or the code throws an exception.

- `setTestGetFeedElementsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, filter, result)`
  - Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElementsFromFeed` is called with matching parameters in a test context. Use the `getFeedElementsFromFeed` method with the same parameters or the code throws an exception.

- `setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, result)`
  - Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElementsFromFeed` is called with matching parameters in a test context. Use the `getFeedElementsFromFeed` method with the same parameters or the code throws an exception.

- `setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam, result)`
  - Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElementsFromFeed` is called with matching parameters in a test context. Use the `getFeedElementsFromFeed` method with the same parameters or the code throws an exception.

- `setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, result)`
  - Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElementsFromFeed` is called with matching parameters in a test context. Use the `getFeedElementsFromFeed` method with the same parameters or the code throws an exception.

- `setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, showInternalOnly, result)`
  - Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElementsFromFeed` is called with matching parameters in a test context. Use the `getFeedElementsFromFeed` method with the same parameters or the code throws an exception.
Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsFromFeed is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

Register a ConnectApi.FeedElementPage object to be returned when the matching ConnectApi.getFeedElementsFromFeed method is called in a test context. Use the method with the same parameters or you receive an exception.

Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsFromFeed is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsFromFeed is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsFromFeed is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

Register a ConnectApi.FeedElementPage object to be returned when the matching ConnectApi.getFeedElementsFromFeed method is called in a test context. Use the method with the same parameters or you receive an exception.

Register a ConnectApi.FeedElementPage object to be returned when the matching ConnectApi.getFeedElementsFromFilterFeed method is called in a test context. Use the method with the same parameters or the code throws an exception.

Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsFromFilterFeed method is called in a test context. Use the method with the same parameters or the code throws an exception.

Register a ConnectApi.FeedElementPage object to be returned when the matching getFeedElementsFromFilterFeed method is called in a test context. Use the method with the same parameters or the code throws an exception.

Register a ConnectApi.FeedElementPage object to be returned when the matching getFeedElementsFromFilterFeed method is called in a test context. Use the method with the same parameters or the code throws an exception.
setTestGetFeedElementsFromFilterFeedUpdatedSince(communityId, subjectId, keyPrefix, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, result)
Register a ConnectApi.FeedElementPage object to be returned when the getFeedElementsFromFilterFeedUpdatedSince method is called in a test context.

setTestGetFeedElementsUpdatedSince(communityId, feedType, recentCommentCount, density, pageParam, pageSize, updatedSince, result)
Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsUpdatedSince is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

setTestGetFeedElementsUpdatedSince(communityId, feedType, recentCommentCount, density, pageParam, pageSize, updatedSince, filter, result)
Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsUpdatedSince is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

setTestGetFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince, result)
Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsUpdatedSince is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

setTestGetFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince, showInternalOnly, result)
Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsUpdatedSince is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

setTestGetFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, filter, result)
Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsUpdatedSince is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

setTestGetFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, showInternalOnly, filter, result)
Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsUpdatedSince is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

setTestGetFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, showInternalOnly, customFilter, result)
Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsUpdatedSince is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

setTestGetFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, showInternalOnly, customFilter, result)
Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsUpdatedSince is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

setTestGetFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, showInternalOnly, customFilter, result)
Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsUpdatedSince is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

setTestGetFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, showInternalOnly, customFilter, result)
Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsUpdatedSince is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

setTestGetRelatedPosts(communityId, feedElementId, filter, maxResults, result)
Register a ConnectApi.RelatedFeedPosts object to be returned when the matching ConnectApi.getRelatedPosts(communityId, feedElementId, filter, maxResults) method is called in a test context. Use the method with the same parameters or you receive an exception.
setTestGetTopUnansweredQuestions(communityId, result) (Pilot)
Register a ConnectApi.FeedElementPage object to be returned when the matching
ConnectApi.getTopUnansweredQuestions method is called in a test context. Use the method with the same parameters
or you receive an exception.

setTestGetTopUnansweredQuestions(communityId, filter, result) (Pilot)
Register a ConnectApi.FeedElementPage object to be returned when the matching
ConnectApi.getTopUnansweredQuestions method is called in a test context. Use the method with the same parameters
or you receive an exception.

setTestGetTopUnansweredQuestions(communityId, pageSize, result) (Pilot)
Register a ConnectApi.FeedElementPage object to be returned when the matching
ConnectApi.getTopUnansweredQuestions method is called in a test context. Use the method with the same parameters
or you receive an exception.

setTestGetTopUnansweredQuestions(communityId, filter, pageSize, result) (Pilot)
Register a ConnectApi.FeedElementPage object to be returned when the matching
ConnectApi.getTopUnansweredQuestions method is called in a test context. Use the method with the same parameters
or you receive an exception.

setTestSearchFeedElements(communityId, q, result)
Register a ConnectApi.FeedElementPage object to be returned when the matching
ConnectApi.searchFeedElements method is called in a test context. Use the method with the same parameters or you
receive an exception.

setTestSearchFeedElements(communityId, q, sortParam, result)
Register a ConnectApi.FeedElementPage object to be returned when the matching
ConnectApi.searchFeedElements method is called in a test context. Use the method with the same parameters or you
receive an exception.

setTestSearchFeedElements(communityId, q, threadedCommentsCollapsed, result)
Register a ConnectApi.FeedElementPage object to be returned when the matching
ConnectApi.searchFeedElements method is called in a test context. Use the method with the same parameters or you
receive an exception.

setTestSearchFeedElements(communityId, q, pageParam, pageSize, result)
Register a ConnectApi.FeedElementPage object to be returned when the matching
ConnectApi.searchFeedElements method is called in a test context. Use the method with the same parameters or you
receive an exception.

setTestSearchFeedElements(communityId, q, pageParam, pageSize, sortParam, result)
Register a ConnectApi.FeedElementPage object to be returned when the matching
ConnectApi.searchFeedElements method is called in a test context. Use the method with the same parameters or you
receive an exception.

setTestSearchFeedElements(communityId, q, recentCommentCount, pageParam, pageSize, sortParam, result)
Register a ConnectApi.FeedElementPage object to be returned when the matching
ConnectApi.searchFeedElements method is called in a test context. Use the method with the same parameters or you
receive an exception.
setTestSearchFeedElementsInFeed(communityId, feedType, q, result)
Register a ConnectApi.FeedElementPage object to be returned when the matching
ConnectApi.searchFeedElementsInFeed method is called in a test context. Use the method with the same parameters
or you receive an exception.

setTestSearchFeedElementsInFeed(communityId, feedType, pageParam, pageSize, sortParam, q, result)
Register a ConnectApi.FeedElementPage object to be returned when the matching
ConnectApi.searchFeedElementsInFeed method is called in a test context. Use the method with the same parameters
or you receive an exception.

setTestSearchFeedElementsInFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, q, result)
Register a ConnectApi.FeedElementPage object to be returned when the matching
ConnectApi.searchFeedElementsInFeed method is called in a test context. Use the method with the same parameters
or you receive an exception.

setTestSearchFeedElementsInFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, q, filter, result)
Register a ConnectApi.FeedElementPage object to be returned when the matching
ConnectApi.searchFeedElementsInFeed method is called in a test context. Use the method with the same parameters
or you receive an exception.

setTestSearchFeedElementsInFeed(communityId, feedType, subjectId, q, result)
Register a ConnectApi.FeedElementPage object to be returned when the matching
ConnectApi.searchFeedElementsInFeed method is called in a test context. Use the method with the same parameters
or you receive an exception.

setTestSearchFeedElementsInFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam, q, result)
Register a ConnectApi.FeedElementPage object to be returned when the matching
ConnectApi.searchFeedElementsInFeed method is called in a test context. Use the method with the same parameters
or you receive an exception.

setTestSearchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, result)
Register a ConnectApi.FeedElementPage object to be returned when the matching
ConnectApi.searchFeedElementsInFeed method is called in a test context. Use the method with the same parameters
or you receive an exception.

setTestSearchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, filter, result)
Register a ConnectApi.FeedElementPage object to be returned when the matching
ConnectApi.searchFeedElementsInFeed method is called in a test context. Use the method with the same parameters
or the code throws an exception.

setTestSearchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, customFilter, result)
Register a ConnectApi.FeedElementPage object to be returned when the matching
ConnectApi.searchFeedElementsInFeed method is called in a test context. Use the method with the same parameters
or you receive an exception.

setTestSearchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, showInternalOnly, result)
Register a ConnectApi.FeedElementPage object to be returned when the matching
ConnectApi.searchFeedElementsInFeed method is called in a test context. Use the method with the same parameters
or you receive an exception.
setTestSearchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, showInternalOnly, filter, result)

Register a ConnectApi.FeedElementPage object to be returned when the matching ConnectApi.searchFeedElementsInFeed method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, showInternalOnly, customFilter, result)

Register a ConnectApi.FeedElementPage object to be returned when the matching ConnectApi.searchFeedElementsInFeed method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchFeedElementsInFilterFeed(communityId, subjectId, keyPrefix, q, result)

Register a ConnectApi.FeedElementPage object to be returned when the matching ConnectApi.searchFeedElementsInFilterFeed method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchFeedElementsInFilterFeed(communityId, subjectId, keyPrefix, pageParam, pageSize, sortParam, q, result)

Register a ConnectApi.FeedElementPage object to be returned when the matching ConnectApi.searchFeedElementsInFilterFeed method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchFeedElementsInFilterFeed(communityId, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, sortParam, q, result)

Register a ConnectApi.FeedElementPage object to be returned when the matching ConnectApi.searchFeedElementsInFilterFeed method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchStreams(communityId, q, result)

Register a ConnectApi.ChatterStreamPage object to be returned when the matching ConnectApi.searchStream(communityId, q) method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchStreams(communityId, q, sortParam, result)

Register a ConnectApi.ChatterStreamPage object to be returned when the matching ConnectApi.searchStream(communityId, q, sortParam) method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchStreams(communityId, q, pageParam, pageSize, result)

Register a ConnectApi.ChatterStreamPage object to be returned when the matching ConnectApi.searchStreams(communityId, q, pageParam, pageSize) method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchStreams(communityId, q, pageParam, pageSize, sortParam, result)

Register a ConnectApi.ChatterStreamPage object to be returned when the matching ConnectApi.searchStreams(communityId, q, pageParam, pageSize, sortParam) method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchStreams(communityId, q, pageParam, pageSize, sortParam, globalScope, result)

Register a ConnectApi.ChatterStreamPage object to be returned when the matching ConnectApi.searchStreams(communityId, q, pageParam, pageSize, sortParam, globalScope) method is called in a test context. Use the method with the same parameters or you receive an exception.
setTestGetFeedElementsFromFeed(communityId, feedType, result)

Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsFromFeed is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

API Version
31.0

Signature
public static Void setTestGetFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, ConnectApi.FeedElementPage result)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Type of feed. Valid values are Company, DirectMessageModeration, DirectMessages, Home, Moderation, and PendingReview.

result
Type: ConnectApi.FeedElementPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
getFeedElementsFromFeed(communityId, feedType)

setTestGetFeedElementsFromFeed(communityId, feedType, pageParam, pageSize, sortParam, result)

Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsFromFeed is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

API Version
31.0
Signature

public static Void setTestGetFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedElementPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
The only valid value for this parameter is Company.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
If you pass in null, the default value CreatedDateDesc is used.

result
Type: ConnectApi.FeedElementPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
getFeedElementsFromFeed(communityId, feedType, pageParam, pageSize, sortParam)
**setTestGetFeedElementsFromFeed**

`setTestGetFeedElementsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, result)`

Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElementsFromFeed` is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

**API Version**

31.0

**Signature**

```java
public static Void setTestGetFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrders sortParam, ConnectApi.FeedElementPage result)
```

**Parameters**

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **feedType**
  - Type: `ConnectApi.FeedType`
  - Type of feed. Valid values are `Company`, `DirectMessageModeration`, `DirectMessages`, `Home`, `Moderation`, and `PendingReview`.

- **recentCommentCount**
  - Type: `Integer`
  - Maximum number of comments to return with each feed element. The default value is 3.

- **density**
  - Type: `ConnectApi.FeedDensity`
  - Specify the amount of content in a feed.
    - `AllUpdates`—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
    - `FewerUpdates`—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

- **pageParam**
  - Type: `String`
  - Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

- **pageSize**
  - Type: `Integer`
  - Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

- **sortParam**
  - Type: `ConnectApi.FeedSortOrder`
  - Values are:
• CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
• CreatedDateDesc—Sorts by most recent creation date.
• LastModifiedDateDesc—Sorts by most recent activity.
• MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
• Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

If you pass in `null`, the default value CreatedDateDesc is used.

result
 Type: ConnectApi.FeedElementPage
 Object containing test data.

Return Value
 Type: Void

SEE ALSO:
- `getFeedElementsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam)`

`setTestGetFeedElementsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, filter, result)`

Register a ConnectApi.FeedElementPage object to be returned when `getFeedElementsFromFeed` is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

API Version
32.0

Signature

`public static Void setTestGetFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedFilter filter, ConnectApi.FeedElementPage result)`

Parameters

- `communityId`
  Type: String
  ID for an Experience Cloud site, internal, or `null`.

- `feedType`
  Type: ConnectApi.FeedType
  The type of feed. The only valid value is Home.
recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
If you pass in null, the default value CreatedDateDesc is used.

filter
Type: ConnectApi.FeedFilter
Specifies the feed filters.
- AllQuestions—Feed elements that are questions.
- AuthoredBy—Feed elements authored by the user profile owner. This value is valid only for the UserProfile feed.
- CommunityScoped—Feed elements that are scoped to Experience Cloud sites. Currently, these feed elements have a User or a Group parent record. However, other parent record types could be scoped to sites in the future. Feed elements that are always visible in all sites are filtered out. This value is valid only for the UserProfile feed.
- QuestionsWithCandidateAnswers—Feed elements that are questions that have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
- QuestionsWithCandidateAnswersReviewedPublished—Feed elements that are questions that have candidate answers that have been reviewed or published. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **Read**—Feed elements that are older than 30 days or are marked as read for the context user. Includes existing feed elements when the context user joined the group. This value is valid only for the Record feed of a group.
- **SolvedQuestions**—Feed elements that are questions and that have a best answer.
- **UnansweredQuestions**—Feed elements that are questions and that don’t have any answers.
- **UnansweredQuestionsWithCandidateAnswers**—Feed elements that are questions that don’t have answers but have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **Unread**—Feed elements that are created in the past 30 days and aren’t marked as read for the context user. This value is valid only for the Record feed of a group.
- **UnsolvedQuestions**—Feed elements that are questions and that don’t have a best answer.

`result`

Type: `ConnectApi.FeedElementPage`  
Object containing test data.

**Return Value**

Type: `Void`

**SEE ALSO:**  
`getFeedElementsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, filter)`  
*Apex Developer Guide: Testing ConnectApi Code*

`setTestGetFeedElementsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, filter, threadedCommentsCollapsed, result)`

Register a `ConnectApi.FeedElementPage` object to be returned when the matching `ConnectApi.getFeedElementsFromFeed` method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**

44.0

**Signature**

```java
public static Void setTestGetFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedFilter filter, Boolean threadedCommentsCollapsed, ConnectApi.FeedElementPage result)
```

**Parameters**

- `communityId`
  
  Type: `String`  
  ID for an Experience Cloud site, `internal`, or `null`.  

667
**feedType**
Type: `ConnectApi.FeedType`
The type of feed. The only valid value is `Home`.

**recentCommentCount**
Type: `Integer`
Maximum number of comments to return with each feed item. The default value is 3.

**density**
Type: `ConnectApi.FeedDensity`
Specify the amount of content in a feed.
- **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**
Type: `String`
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**
Type: `Integer`
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**
Type: `ConnectApi.FeedSortOrder`
Values are:
- **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- **Relevance**—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.
If you pass in `null`, the default value `CreatedDateDesc` is used.

**filter**
Type: `ConnectApi.FeedFilter`
Specifies the feed filters.
- **AllQuestions**—Feed elements that are questions.
- **AuthoredBy**—Feed elements authored by the user profile owner. This value is valid only for the `UserProfile` feed.
- **CommunityScoped**—Feed elements that are scoped to Experience Cloud sites. Currently, these feed elements have a `User` or a `Group` parent record. However, other parent record types could be scoped to sites in the future. Feed elements that are always visible in all sites are filtered out. This value is valid only for the `UserProfile` feed.
- **QuestionsWithCandidateAnswers**—Feed elements that are questions that have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
• QuestionsWithCandidateAnswersReviewedPublished—Feed elements that are questions that have candidate answers that have been reviewed or published. This value is valid only for users with the Access Einstein-Generated Answers permission.

• Read—Feed elements that are older than 30 days or are marked as read for the context user. Includes existing feed elements when the context user joined the group. This value is valid only for the Record feed of a group.

• SolvedQuestions—Feed elements that are questions and that have a best answer.

• UnansweredQuestions—Feed elements that are questions and that don’t have any answers.

• UnansweredQuestionsWithCandidateAnswers—Feed elements that are questions that don’t have answers but have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.

• Unread—Feed elements that are created in the past 30 days and aren’t marked as read for the context user. This value is valid only for the Record feed of a group.

• UnsolvedQuestions—Feed elements that are questions and that don’t have a best answer.

threadedCommentsCollapsed
Type: Boolean
Specifies whether to return threaded comments in a collapsed style (true) or not (false). If you pass in null, the default is false.

result
Type: ConnectApi.FeedElementPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
getFeedElementsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, filter, threadedCommentsCollapsed)

setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, result)
Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsFromFeed is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

API Version
31.0

Signature
public static Void setTestGetFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, ConnectApi.FeedElementPage result)
Parameters

`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

`feedType`
Type: `ConnectApi.FeedType`
The feed type.

`subjectId`
Type: `String`
ID of the context user or the alias `me`.

`result`
Type: `ConnectApi.FeedElementPage`
Object containing test data.

Return Value
Type: `Void`

SEE ALSO:
- `getFeedElementsFromFeed(communityId, feedType, subjectId)`

`setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam, result)`
Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElementsFromFeed` is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

API Version
31.0

Signature
```
public static Void setTestGetFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedElementPage result)
```

Parameters

`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

`feedType`
Type: `ConnectApi.FeedType`
Type of feed. Valid values include every ConnectApi.FeedType except Company, DirectMessageModeration, DirectMessages, Filter, Home, Landing, Moderation, and PendingReview.

`subjectId`

Type: String

If `feedType` is Record, `subjectId` can be any record ID, including a group ID. If `feedType` is Streams, `subjectId` must be a stream ID. If `feedType` is Topics, `subjectId` must be a topic ID. If `feedType` is UserProfile, `subjectId` can be any user ID. If the `feedType` is any other value, `subjectId` must be the ID of the context user or the alias `me`.

`pageParam`

Type: String

Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in null, the first page is returned.

`pageSize`

Type: Integer

Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

`sortParam`

Type: ConnectApi.FeedSortOrder

Values are:

- `CreatedDateAsc`—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- `CreatedDateDesc`—Sorts by most recent creation date.
- `LastModifiedDateDesc`—Sorts by most recent activity.
- `MostViewed`—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- `Relevance`—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

If you pass in null, the default value `CreatedDateDesc` is used.

`result`

Type: ConnectApi.FeedElementPage

Object containing test data.

Return Value

Type: Void

SEE ALSO:

- `getFeedElementsFromFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam)`
- *Apex Developer Guide: Testing ConnectApi Code*

`setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, result)`

Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElementsFromFeed` is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.
API Version
31.0

Signature
public static Void setTestGetFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedElementPage result)

Parameters
communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

feedType
  Type: ConnectApi.FeedType
  Type of feed. Valid values include every ConnectApi.FeedType except Company, DirectMessageModeration, DirectMessages, Filter, Home, Landing, Moderation, and PendingReview.

subjectId
  Type: String
  If feedType is Record, subjectId can be any record ID, including a group ID. If feedType is Streams, subjectId must be a stream ID. If feedType is Topics, subjectId must be a topic ID. If feedType is UserProfile, subjectId can be any user ID. If the feedType is any other value, subjectId must be the ID of the context user or the alias me.

recentCommentCount
  Type: Integer
  Maximum number of comments to return with each feed element. The default value is 3.

density
  Type: ConnectApi.FeedDensity
  Specify the amount of content in a feed.
  - AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
  - FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
  Type: String
  Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
  Type: Integer
  Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
  Type: ConnectApi.FeedSortOrder
Values are:

- **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- **Relevance**—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

If you pass in `null`, the default value CreatedDateDesc is used.

`result`

Type: `ConnectApi.FeedElementPage`

Object containing test data.

**Return Value**

Type: Void

**SEE ALSO:**

getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam)


**setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, showInternalOnly, result)**

Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElementsFromFeed` is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

**API Version**

31.0

**Signature**

```java
public static Void setTestGetFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, Boolean showInternalOnly, ConnectApi.FeedElementPage result)
```

**Parameters**

- **communityId**
  
  Type: `String`
  
  ID for an Experience Cloud site, internal, or `null`.

- **feedType**
  
  Type: `ConnectApi.FeedType`
Value must be `ConnectApi.FeedType.Record`.

**subjectId**
Type: `String`
Any record ID, including a group ID.

**recentCommentCount**
Type: `Integer`
Maximum number of comments to return with each feed item. The default value is 3.

**density**
Type: `ConnectApi.FeedDensity`
Specify the amount of content in a feed.
- **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**
Type: `String`
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**
Type: `Integer`
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**
Type: `ConnectApi.FeedSortOrder`
Values are:
- **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for Home feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- **Relevance**—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

If you pass in `null`, the default value `CreatedDateDesc` is used.

**showInternalOnly**
Type: `Boolean`
Specifies whether to show only feed items from internal (non-Experience Cloud site) users (`true`), or not (`false`). The default value is `false`.

**result**
Type: `ConnectApi.FeedElementPage`
Object containing test data.
Return Value
Type: Void

SEE ALSO:
- getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, showInternalOnly)

setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, filter, result)

Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsFromFeed is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

API Version
35.0

Signature
public static Void setTestGetFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedFilter filter, ConnectApi.FeedElementPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Value must be ConnectApi.FeedType.UserProfile.

subjectId
Type: String
ID of any user. To specify the context user, use the user ID or the alias me.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
• FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**

Type: String

Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

**pageSize**

Type: Integer

Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

**sortParam**

Type: ConnectApi.FeedSortOrder

Values are:

- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

If you pass in null, the default value CreatedDateDesc is used.

**filter**

Type: ConnectApi.FeedFilter

Value must be ConnectApi.FeedFilter.CommunityScoped. Filters the feed to include only feed elements that are scoped to Experience Cloud sites. Feed elements that are always visible in all sites are filtered out. Currently, feed elements scoped to sites have a User or a Group parent record. However, other parent record types could be scoped to sites in the future.

**result**

Type: ConnectApi.FeedElementPage

Object containing test data.

**Return Value**

Type: Void

SEE ALSO:

getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, filter)

*Apex Developer Guide: Testing ConnectApi Code*
setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, filter, threadedCommentsCollapsed, result)

Register a `ConnectApi.FeedElementPage` object to be returned when the matching `ConnectApi.getFeedElementsFromFeed` method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**

44.0

**Signature**

```java
public static Void setTestGetFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedFilter filter, Boolean threadedCommentsCollapsed, ConnectApi.FeedElementPage result)
```

**Parameters**

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **feedType**
  - Type: `ConnectApi.FeedType`
  - Value must be `ConnectApi.FeedType.UserProfile`.

- **subjectId**
  - Type: `String`
  - ID of any user. To specify the context user, use the user ID or the alias `me`.

- **recentCommentCount**
  - Type: `Integer`
  - Maximum number of comments to return with each feed element. The default value is 3.

- **density**
  - Type: `ConnectApi.FeedDensity`
  - Specify the amount of content in a feed.
    - `AllUpdates`—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
    - `FewerUpdates`—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

- **pageParam**
  - Type: `String`
  - Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.
**pageSize**
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

**sortParam**
Type: `ConnectApi.FeedSortOrder`
Values are:
- **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for Home feeds when the `ConnectApi.FeedFilter` is UnansweredQuestions.
- **Relevance**—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

If you pass in null, the default value **CreatedDateDesc** is used.

**filter**
Type: `ConnectApi.FeedFilter`
Value must be `ConnectApi.FeedFilter.CommunityScoped`. Filters the feed to include only feed elements that are scoped to Experience Cloud sites. Feed elements that are always visible in all sites are filtered out. Currently, feed elements scoped to sites have a User or a Group parent record. However, other parent record types could be scoped to sites in the future.

**threadedCommentsCollapsed**
Type: Boolean
Specifies whether to return threaded comments in a collapsed style (true) or not (false). If you pass in null, the default is false.

**result**
Type: `ConnectApi.FeedElementPage`
Object containing test data.

**Return Value**
Type: Void

**SEE ALSO:**
* `getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, filter, threadedCommentsCollapsed)`

```java
setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, customFilter, result)
```

Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElementsFromFeed` is called with matching parameters in a test context. Use the `get feed` method with the same parameters or the code throws an exception.
API Version

40.0

Signature

```java
public static Void setTestGetFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String customFilter, ConnectApi.FeedElementPage result)
```

Parameters

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- **feedType**
  - Type: ConnectApi.FeedType
  - Value must be ConnectApi.FeedType.Record.

- **subjectId**
  - Type: String
  - The ID of a case.

- **recentCommentCount**
  - Type: Integer
  - Maximum number of comments to return with each feed element. The default value is 3.

- **density**
  - Type: ConnectApi.FeedDensity
  - Specify the amount of content in a feed.
    - AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
    - FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

- **pageParam**
  - Type: String
  - Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

- **pageSize**
  - Type: Integer
  - Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

- **sortParam**
  - Type: ConnectApi.FeedSortOrder
  - Values are:
    - CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- **Relevance**—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

If you pass in **null**, the default value CreatedDateDesc is used.

`customFilter`

Type: **String**

Custom filter that applies only to the case feed. See `customFeedFilter` in the *Metadata API Developer Guide* for supported values.

`result`

Type: **ConnectApi.FeedElementPage**

Object containing test data.

**Return Value**

Type: Void

SEE ALSO:

- `getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, customFilter)`
- *Apex Developer Guide: Testing ConnectApi Code*

`setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly, result)`

Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElementsFromFeed` is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

**API Version**

31.0

**Signature**

```
public static Void setTestGetFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, Integer elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrdersortParam, Boolean showInternalOnly, ConnectApi.FeedElementPage result)
```

**Parameters**

`communityId`

Type: **String**

ID for an Experience Cloud site, internal, or **null**.
**feedType**
Type: `ConnectApi.FeedType`
Value must be `ConnectApi.FeedType.Record`.

**subjectId**
Type: `String`
Any record ID, including a group ID.

**recentCommentCount**
Type: `Integer`
Maximum number of comments to return with each feed item. The default value is 3.

**elementsPerBundle**
Type: `Integer`
Maximum number of feed elements per bundle. The default and maximum value is 10.

**density**
Type: `ConnectApi.FeedDensity`
Specify the amount of content in a feed.
- **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**
Type: `String`
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**
Type: `Integer`
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**
Type: `ConnectApi.FeedSortOrder`
Values are:
- **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- **Relevance**—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.
If you pass in `null`, the default value `CreatedDateDesc` is used.

**showInternalOnly**
Type: `Boolean`
Specifies whether to show only feed items from internal (non-Experience Cloud site) users (`true`), or not (`false`). The default value is `false`. 
result

Type: `ConnectApi.FeedElementPage`

Object containing test data.

Return Value

Type: Void

SEE ALSO:

- `getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly)`

```java
public static void setTestGetFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, Integer elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, Boolean showInternalOnly, ConnectApi.FeedFilter filter, ConnectApi.FeedElementPage result)
```

Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElementsFromFeed` is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

API Version

32.0

Signature

```java
public static void setTestGetFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, Integer elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, Boolean showInternalOnly, ConnectApi.FeedFilter filter, ConnectApi.FeedElementPage result)
```

Parameters

- `communityId`
  Type: `String`
  ID for an Experience Cloud site, `internal`, or `null`.

- `feedType`
  Type: `ConnectApi.FeedType`
  Value must be `ConnectApi.FeedType.Record`.

- `subjectId`
  Type: `String`
  Any record ID, including a group ID.

- `recentCommentCount`
  Type: `Integer`
  Maximum number of comments to return with each feed item. The default value is 3.
elementsPerBundle
Type: Integer
Maximum number of feed elements per bundle. The default and maximum value is 10.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
If you pass in null, the default value CreatedDateDesc is used.

showInternalOnly
Type: Boolean
Specifies whether to show only feed items from internal (non-Experience Cloud site) users (true), or not (false). The default value is false.

filter
Type: ConnectApi.FeedFilter
Specifies the feed filters.
- AllQuestions—Feed elements that are questions.
- AuthoredBy—Feed elements authored by the user profile owner. This value is valid only for the UserProfile feed.
- CommunityScoped—Feed elements that are scoped to Experience Cloud sites. Currently, these feed elements have a User or a Group parent record. However, other parent record types could be scoped to sites in the future. Feed elements that are always visible in all sites are filtered out. This value is valid only for the UserProfile feed.
ChatterFeeds Class

- **QuestionsWithCandidateAnswers**—Feed elements that are questions that have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **QuestionsWithCandidateAnswersReviewedPublished**—Feed elements that are questions that have candidate answers that have been reviewed or published. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **Read**—Feed elements that are older than 30 days or are marked as read for the context user. Includes existing feed elements when the context user joined the group. This value is valid only for the Record feed of a group.
- **SolvedQuestions**—Feed elements that are questions and that have a best answer.
- **UnansweredQuestions**—Feed elements that are questions and that don’t have any answers.
- **UnansweredQuestionsWithCandidateAnswers**—Feed elements that are questions that don’t have answers but have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **Unread**—Feed elements that are created in the past 30 days and aren’t marked as read for the context user. This value is valid only for the Record feed of a group.
- **UnsolvedQuestions**—Feed elements that are questions and that don’t have a best answer.

result

Type: `ConnectApi.FeedElementPage`

Object containing test data.

**Return Value**

Type: Void

**SEE ALSO:**

- `getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly, filter)`
- *Apex Developer Guide: Testing ConnectApi Code*

`setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly, filter, threadedCommentsCollapsed, result)`

Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElementsFromFeed` is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

**API Version**

44.0

**Signature**

```java
public static Void setTestGetFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, Integer elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, Boolean showInternalOnly, ConnectApi.FeedFilter filter, Boolean threadedCommentsCollapsed, ConnectApi.FeedElementPage result)
```
Parameters

`communityId`  
Type: String  
ID for an Experience Cloud site, internal, or null.

`feedType`  
Type: `ConnectApi.FeedType`  
Value must be `ConnectApi.FeedType.Record`.

`subjectId`  
Type: String  
Any record ID, including a group ID.

`recentCommentCount`  
Type: Integer  
Maximum number of comments to return with each feed item. The default value is 3.

`elementsPerBundle`  
Type: Integer  
Maximum number of feed elements per bundle. The default and maximum value is 10.

`density`  
Type: `ConnectApi.FeedDensity`  
Specify the amount of content in a feed.
- **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

`pageParam`  
Type: String  
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

`pageSize`  
Type: Integer  
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

`sortParam`  
Type: `ConnectApi.FeedSortOrder`  
Values are:
- **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for Home feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- **Relevance**—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
If you pass in `null`, the default value `CreatedDateDesc` is used.

`showInternalOnly`
- **Type:** `Boolean`
- Specifies whether to show only feed items from internal (non-Experience Cloud site) users (true), or not (false). The default value is `false`.

`filter`
- **Type:** `ConnectApi.FeedFilter`
- Specifies the feed filters.
  - `AllQuestions` — Feed elements that are questions.
  - `AuthoredBy` — Feed elements authored by the user profile owner. This value is valid only for the `UserProfile` feed.
  - `CommunityScoped` — Feed elements that are scoped to Experience Cloud sites. Currently, these feed elements have a User or a Group parent record. However, other parent record types could be scoped to sites in the future. Feed elements that are always visible in all sites are filtered out. This value is valid only for the `UserProfile` feed.
  - `QuestionsWithCandidateAnswers` — Feed elements that are questions that have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
  - `QuestionsWithCandidateAnswersReviewedPublished` — Feed elements that are questions that have candidate answers that have been reviewed or published. This value is valid only for users with the Access Einstein-Generated Answers permission.
  - `Read` — Feed elements that are older than 30 days or are marked as read for the context user. Includes existing feed elements when the context user joined the group. This value is valid only for the `Record` feed of a group.
  - `SolvedQuestions` — Feed elements that are questions and that have a best answer.
  - `UnansweredQuestions` — Feed elements that are questions and that don’t have any answers.
  - `UnansweredQuestionsWithCandidateAnswers` — Feed elements that are questions that don’t have answers but have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
  - `Unread` — Feed elements that are created in the past 30 days and aren’t marked as read for the context user. This value is valid only for the `Record` feed of a group.
  - `UnsolvedQuestions` — Feed elements that are questions and that don’t have a best answer.

`threadedCommentsCollapsed`
- **Type:** `Boolean`
- Specifies whether to return threaded comments in a collapsed style (true) or not (false). If you pass in `null`, the default is `false`.

`result`
- **Type:** `ConnectApi.FeedElementPage`
- Object containing test data.
Return Value

Type: Void

SEE ALSO:

`getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly, filter, threadedCommentsCollapsed)`

*Apex Developer Guide: Testing ConnectApi Code*

**setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly, customFilter, result)**

Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElementsFromFeed` is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

**API Version**

40.0

**Signature**

```java
public static Void setTestGetFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, Integer elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, Boolean showInternalOnly, String customFilter, ConnectApi.FeedElementPage result)
```

**Parameters**

- `communityId`  
  Type: String  
  ID for an Experience Cloud site, internal, or null.

- `feedType`  
  Type: `ConnectApi.FeedType`  
  Value must be `ConnectApi.FeedType.Record`.

- `subjectId`  
  Type: String  
  The ID of a case.

- `recentCommentCount`  
  Type: Integer  
  Maximum number of comments to return with each feed item. The default value is 3.

- `elementsPerBundle`  
  Type: Integer  
  Maximum number of feed elements per bundle. The default and maximum value is 10.
density
  Type: ConnectApi.FeedDensity
  Specify the amount of content in a feed.
  • AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
  • FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
  Type: String
  Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
  Type: Integer
  Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
  Type: ConnectApi.FeedSortOrder
  Values are:
  • CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
  • CreatedDateDesc—Sorts by most recent creation date.
  • LastModifiedDateDesc—Sorts by most recent activity.
  • MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
  • Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
  If you pass in null, the default value CreatedDateDesc is used.

showInternalOnly
  Type: Boolean
  Specifies whether to show only feed items from internal (non-Experience Cloud site) users (true), or not (false). The default value is false.

customFilter
  Type: String
  Custom filter that applies only to the case feed. See customFeedFilter in the Metadata API Developer Guide for supported values.

result
  Type: ConnectApi.FeedElementPage
  Object containing test data.
Return Value
Type: Void

SEE ALSO:
getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly, customFilter)

setTestGetFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly, customFilter, threadedCommentsCollapsed, result)

Register a ConnectApi.FeedElementPage object to be returned when the matching ConnectApi.getFeedElementsFromFeed method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
44.0

Signature
public static Void setTestGetFeedElementsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, Integer elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, Boolean showInternalOnly, String customFilter, Boolean threadedCommentsCollapsed, ConnectApi.FeedElementPage result)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Value must be ConnectApi.FeedType.Record.

subjectId
Type: String
The ID of a case.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed item. The default value is 3.

elementsPerBundle
Type: Integer
Maximum number of feed elements per bundle. The default and maximum value is 10.
density
Type: `ConnectApi.FeedDensity`
Specify the amount of content in a feed.
- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: `String`
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

pageSize
Type: `Integer`
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

sortParam
Type: `ConnectApi.FeedSortOrder`
Values are:
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- Relevance—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.
If you pass in `null`, the default value `CreateDateDesc` is used.

showInternalOnly
Type: `Boolean`
Specifies whether to show only feed items from internal (non-Experience Cloud site) users (`true`), or not (`false`). The default value is `false`.

customFilter
Type: `String`
Custom filter that applies only to the case feed. See `customFeedFilter` in the `Metadata API Developer Guide` for supported values.

threadedCommentsCollapsed
Type: `Boolean`
Specifies whether to return threaded comments in a collapsed style (`true`) or not (`false`). If you pass in `null`, the default is `false`.

result
Type: `ConnectApi.FeedElementPage`
Object containing test data.
Return Value
Type: Void

SEE ALSO:


`getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, showInternalOnly, customFilter, threadedCommentsCollapsed)`

`setTestGetFeedElementsFromFilterFeed(communityId, subjectId, keyPrefix, result)`

A `ConnectApi.FeedElementPage` object to be returned when the matching `getFeedElementsFromFilterFeed` method is called in a test context. Use the method with the same parameters or the code throws an exception.

API Version
31.0

Signature

```java
public static Void setTestGetFeedElementsFromFilterFeed(String communityId, String subjectId, String keyPrefix, ConnectApi.FeedElementPage result)
```

Parameters

- `communityId` Type: `String`
  
  ID for an Experience Cloud site, `internal`, or `null`.

- `subjectId` Type: `String`
  
  ID of the context user or the alias `me`.

- `keyPrefix` Type: `String`
  
  A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

- `result` Type: `ConnectApi.FeedElementPage`
  
  Object containing test data.
Return Value

Type: Void

SEE ALSO:
- `getFeedElementsFromFilterFeed(communityId, subjectId, keyPrefix)`
  *Apex Developer Guide: Testing ConnectApi Code*

`setTestGetFeedElementsFromFilterFeed(communityId, subjectId, keyPrefix, pageParam, pageSize, sortParam, result)`

Register a `ConnectApi.FeedElementPage` object to be returned when the matching `getFeedElementsFromFilterFeed` method is called in a test context. Use the method with the same parameters or the code throws an exception.

API Version

31.0

Signature

```java
public static Void setTestGetFeedElementsFromFilterFeed(String communityId, String subjectId, String keyPrefix, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedElementPage result)
```

Parameters

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, `internal`, or `null`.

- **subjectId**
  - Type: `String`
  - ID of the context user or the alias `me`.

- **keyPrefix**
  - Type: `String`
  - A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of `005` and Group objects have a prefix of `0F9`.

- **pageParam**
  - Type: `String`
  - Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

- **pageSize**
  - Type: `Integer`
  - Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

- **sortParam**
  - Type: `ConnectApi.FeedSortOrder`
Values are:

- **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- **Relevance**—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

If you pass in `null`, the default value CreatedDateDesc is used.

```java
result
Type: ConnectApi.FeedElementPage
Object containing test data.
```

Return Value

Type: Void

SEE ALSO:

- `getFeedElementsFromFilterFeed(communityId, subjectId, keyPrefix, pageParam, pageSize, sortParam)`
- *Apex Developer Guide: Testing ConnectApi Code*

```java
setTestGetFeedElementsFromFilterFeed(communityId, subjectId, keyPrefix, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam, result)
```

Register a `ConnectApi.FeedElementPage` object to be returned when the matching `getFeedElementsFromFilterFeed` method is called in a test context. Use the method with the same parameters or the code throws an exception.

API Version

31.0

Signature

```java
public static Void setTestGetFeedElementsFromFilterFeed(String communityId, String subjectId, String keyPrefix, Integer recentCommentCount, Integer elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedElementPage result)
```

Parameters

- **communityId**
  Type: String
  ID for an Experience Cloud site, internal, or null.

- **subjectId**
  Type: String
ID of the context user or the alias me.

keyPrefix
Type: String
A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

elementsPerBundle
Type: Integer
Maximum number of feed elements per bundle. The default and maximum value is 10.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
• AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
• FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
• CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
• CreatedDateDesc—Sorts by most recent creation date.
• LastModifiedDateDesc—Sorts by most recent activity.
• MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
• Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
If you pass in null, the default value CreatedDateDesc is used.

result
Type: ConnectApi.FeedElementPage
Object containing test data.
Return Value
Type: Void

SEE ALSO:
getFeedElementsFromFilterFeed(communityId, subjectId, keyPrefix, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam)


setTestGetFeedElementsFromFilterFeedUpdatedSince(communityId, subjectId, keyPrefix, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, result)

Register a ConnectApi.FeedElementPage object to be returned when the getFeedElementsFromFilterFeedUpdatedSince method is called in a test context.

API Version
31.0

Signature
public static Void setTestGetFeedElementsFromFilterFeedUpdatedSince(String communityId, String subjectId, String keyPrefix, Integer recentCommentCount, Integer elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, String updatedSince, ConnectApi.FeedElementPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subjectId
Type: String
ID of the context user or the alias me.

keyPrefix
Type: String
A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

elementsPerBundle
Type: Integer
Maximum number of feed elements per bundle. The default and maximum value is 10.
Density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
  • AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
  • FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

PageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

PageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

UpdatedSince
Type: String
Opaque token defining the modification timestamp of the feed and the sort order.
The updatedSince parameter doesn’t return feed elements that are created in the same second as the call.

Result
Type: ConnectApi.FeedElementPage
Object containing test data.

Return Value
Type: Void

See Also:
  getFeedElementsFromFilterFeedUpdatedSince(communityId, subjectId, keyPrefix, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince)

setTestGetFeedElementsUpdatedSince(communityId, feedType, recentCommentCount, density, pageParam, pageSize, updatedSince, result)
Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsUpdatedSince is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

API Version
31.0
Signature

```java
public static Void setTestGetFeedElementsUpdatedSince(String communityId,
ConnectApi.FeedType feedType, Integer recentCommentCount, ConnectApi.FeedDensity density,
String pageParam, Integer pageSize, String updatedSince, ConnectApi.FeedElementPage result)
```

Parameters

**communityId**
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

**feedType**
Type: `ConnectApi.FeedType`
Type of feed. Valid values are `Company`, `DirectMessageModeration`, `Home`, `Moderation`, and `PendingReview`.

**recentCommentCount**
Type: `Integer`
Maximum number of comments to return with each feed element. The default value is 3.

**density**
Type: `ConnectApi.FeedDensity`
Specify the amount of content in a feed.
- **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**
Type: `String`
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**
Type: `Integer`
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**updatedSince**
Type: `String`
An opaque token containing information about the last modified date of the feed. Do not construct this token. Retrieve this token from the `updatesToken` property of the `ConnectApi.FeedElementPage` response body.

The `updatedSince` parameter doesn’t return feed elements that are created in the same second as the call.

**result**
Type: `ConnectApi.FeedElementPage`
Object containing test data.
Return Value
Type: Void

SEE ALSO:
getFeedElementsUpdatedSince(communityId, feedType, recentCommentCount, density, pageParam, pageSize, updatedSince)

setTestGetFeedElementsUpdatedSince(communityId, feedType, recentCommentCount, density, pageParam, pageSize, updatedSince, filter, result)

Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsUpdatedSince is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

API Version
32.0

Signature
public static Void setTestGetFeedElementsUpdatedSince(String communityId, ConnectApi.FeedType feedType, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, String updatedSince, ConnectApi.FeedFilter filter, ConnectApi.FeedElementPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Type of feed. Valid values are Company, DirectMessageModeration, DirectMessages, Home, Moderation, and PendingReview.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.

• AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
• FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

### pageSize
Type: `Integer`

Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

### updatedSince
Type: `String`

An opaque token containing information about the last modified date of the feed. Do not construct this token. Retrieve this token from the `updatesToken` property of the `ConnectApi.FeedElementPage` response body. The `updatedSince` parameter doesn’t return feed elements that are created in the same second as the call.

### filter
Type: `ConnectApi.FeedFilter`

Specifies the feed filters.

- **AllQuestions**—Feed elements that are questions.
- **AuthoredBy**—Feed elements authored by the user profile owner. This value is valid only for the `UserProfile` feed.
- **CommunityScoped**—Feed elements that are scoped to Experience Cloud sites. Currently, these feed elements have a User or a Group parent record. However, other parent record types could be scoped to sites in the future. Feed elements that are always visible in all sites are filtered out. This value is valid only for the `UserProfile` feed.
- **QuestionsWithCandidateAnswers**—Feed elements that are questions that have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **QuestionsWithCandidateAnswersReviewedPublished**—Feed elements that are questions that have candidate answers that have been reviewed or published. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **Read**—Feed elements that are older than 30 days or are marked as read for the context user. Includes existing feed elements when the context user joined the group. This value is valid only for the `Record` feed of a group.
- **SolvedQuestions**—Feed elements that are questions and that have a best answer.
- **UnansweredQuestions**—Feed elements that are questions and that don’t have any answers.
- **UnansweredQuestionsWithCandidateAnswers**—Feed elements that are questions that don’t have answers but have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **Unread**—Feed elements that are created in the past 30 days and aren’t marked as read for the context user. This value is valid only for the `Record` feed of a group.
- **UnsolvedQuestions**—Feed elements that are questions and that don’t have a best answer.

### result
Type: `ConnectApi.FeedElementPage`

Object containing test data.
Return Value
Type: Void

SEE ALSO:
getFeedElementsUpdatedSince(communityId, feedType, recentCommentCount, density, pageParam, pageSize, updatedSince, filter)

setTestGetFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince, result)

Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsUpdatedSince is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

API Version
31.0

Signature
public static Void setTestGetFeedElementsUpdatedSince(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, String updatedSince, ConnectApi.FeedElementPage result)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
One of these values:
• Files
• Groups
• News
• People
• Record

subjectId
Type: String
If feedType is ConnectApi.Record, subjectId can be any record ID, including a group ID. Otherwise, it must be the context user or the alias me.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed item. The default value is 3.
density
    Type: ConnectApi.FeedDensity
    Specify the amount of content in a feed.
    • AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
    • FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
    Type: String
    Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
    Type: Integer
    Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

updatedSince
    Type: String
    An opaque token containing information about the last modified date of the feed. Do not construct this token. Retrieve this token from the updatesToken property of the ConnectApi.FeedElementPage response body.
    The updatedSince parameter doesn’t return feed elements that are created in the same second as the call.

result
    Type: ConnectApi.FeedElementPage
    Object containing test data.

Return Value
    Type: Void

SEE ALSO:
    getFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince)

setTestGetFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince, showInternalOnly, result)
    Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsUpdatedSince is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

API Version
    31.0
public static Void setTestGetFeedElementsUpdatedSince(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, String updatedSince, Boolean showInternalOnly, ConnectApi.FeedElementPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Value must be ConnectApi.FeedType.Record.

subjectId
Type: String
Any record ID, including a group ID.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.

- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

updatedSince
Type: String
An opaque token containing information about the last modified date of the feed. Do not construct this token. Retrieve this token from the updatesToken property of the ConnectApi.FeedElementPage response body.

The updatedSince parameter doesn’t return feed elements that are created in the same second as the call.

showInternalOnly
Type: Boolean
Specifies whether to show only feed elements from internal (non-Experience Cloud site) users (true), or not (false). The default value is false.

result
Type: ConnectApi.FeedElementPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
getFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince, showInternalOnly)

setTestGetFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, filter, result)
Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsUpdatedSince is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

API Version
35.0

Signature
public static Void setTestGetFeedElementsUpdatedSince(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, Integer elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, String updatedSince, ConnectApi.FeedFilter filter, ConnectApi.FeedElementPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Value must be ConnectApi.FeedType.UserProfile.

subjectId
Type: String
ID of any user. To specify the context user, use the user ID or the alias me.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

elementsPerBundle
   Type: Integer
   Maximum number of feed elements per bundle. The default and maximum value is 10.

density
   Type: ConnectApi.FeedDensity
   Specify the amount of content in a feed.
   - AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
   - FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
   Type: String
   Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
   Type: Integer
   Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

updatedSince
   Type: String
   Opaque token defining the modification timestamp of the feed and the sort order.
   The updatedSince parameter doesn’t return feed elements that are created in the same second as the call.

filter
   Type: ConnectApi.FeedFilter
   Value must be ConnectApi.FeedFilter.CommunityScoped. Filters the feed to include only feed elements that are scoped to Experience Cloud sites. Feed elements that are always visible in all sites are filtered out. Currently, feed elements scoped to sites have a User or a Group parent record. However, other parent record types could be scoped to sites in the future.

result
   Type: ConnectApi.FeedElementPage
   Object containing test data.

Return Value
   Type: Void

SEE ALSO:
   getFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, filter)
setTestGetFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, customFilter, result)

Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsUpdatedSince is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

API Version
40.0

Signature
public static Void setTestGetFeedElementsUpdatedSince(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, Integer elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, String updatedSince, String customFilter, ConnectApi.FeedElementPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Value must be ConnectApi.FeedType.Record.

subjectId
Type: String
The ID of a case.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

elementsPerBundle
Type: Integer
Maximum number of feed elements per bundle. The default and maximum value is 10.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

updatedSince
Type: String
Opaque token defining the modification timestamp of the feed and the sort order.
The updatedSince parameter doesn’t return feed elements that are created in the same second as the call.

customFilter
Type: String
Custom filter that applies only to the case feed. See customFeedFilter in the Metadata API Developer Guide for supported values.

result
Type: ConnectApi.FeedElementPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
getFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, customFilter)


setTestGetFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, showInternalOnly, result)

Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsUpdatedSince is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

API Version
31.0

Signature
public static Void setTestGetFeedElementsUpdatedSince(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, Integer elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, String updatedSince, Boolean showInternalOnly, ConnectApi.FeedElementPage result)
Parameters

*communityId*
  Type: String
  ID for an Experience Cloud site, internal, or null.

*feedType*
  Type: `ConnectApi.FeedType`
  Value must be `ConnectApi.FeedType.Record`.

*subjectId*
  Type: String
  Any record ID, including a group ID.

*recentCommentCount*
  Type: Integer
  Maximum number of comments to return with each feed element. The default value is 3.

*elementsPerBundle*
  Type: Integer
  Maximum number of feed elements per bundle. The default and maximum value is 10.

*density*
  Type: `ConnectApi.FeedDensity`
  Specify the amount of content in a feed.
  - AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
  - FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

*pageParam*
  Type: String
  Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

*pageSize*
  Type: Integer
  Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

*updatedSince*
  Type: String
  An opaque token containing information about the last modified date of the feed. Do not construct this token. Retrieve this token from the updatesToken property of the `ConnectApi.FeedElementPage` response body.
  
  The *updatedSince* parameter doesn’t return feed elements that are created in the same second as the call.

*showInternalOnly*
  Type: Boolean
  Specifies whether to show only feed elements from internal (non-Experience Cloud site) users (true), or not (false). The default value is false.

*result*
  Type: `ConnectApi.FeedElementPage`
Object containing test data.

**Return Value**
Type: Void

SEE ALSO:
- `getFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, showInternalOnly)`
- *Apex Developer Guide: Testing ConnectApi Code*

### setTestGetFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, showInternalOnly, filter, result)

Register a `ConnectApi.FeedElementPage` object to be returned when `getFeedElementsUpdatedSince` is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

**API Version**
32.0

**Signature**
```
public static Void setTestGetFeedElementsUpdatedSince(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, Integer elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, String updatedSince, Boolean showInternalOnly, ConnectApi.FeedFilter filter, ConnectApi.FeedElementPage result)
```

**Parameters**
- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or null.
- `feedType`
  Type: `ConnectApi.FeedType`
  Value must be `ConnectApi.FeedType.Record`.
- `subjectId`
  Type: `String`
  Any record ID, including a group ID.
- `recentCommentCount`
  Type: `Integer`
  Maximum number of comments to return with each feed element. The default value is 3.
- `elementsPerBundle`
  Type: `Integer`
Maximum number of feed elements per bundle. The default and maximum value is 10.

**density**
Type: `ConnectApi.FeedDensity`
Specify the amount of content in a feed.
- **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**
Type: `String`
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**
Type: `Integer`
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**updatedSince**
Type: `String`
An opaque token containing information about the last modified date of the feed. Do not construct this token. Retrieve this token from the `updatesToken` property of the `ConnectApi.FeedElementPage` response body.

The `updatedSince` parameter doesn’t return feed elements that are created in the same second as the call.

**showInternalOnly**
Type: `Boolean`
Specifies whether to show only feed elements from internal (non-Experience Cloud site) users (`true`), or not (`false`). The default value is `false`.

**filter**
Type: `ConnectApi.FeedFilter`
Specifies the feed filters.
- **AllQuestions**—Feed elements that are questions.
- **AuthoredBy**—Feed elements authored by the user profile owner. This value is valid only for the `UserProfile` feed.
- **CommunityScoped**—Feed elements that are scoped to Experience Cloud sites. Currently, these feed elements have a User or a Group parent record. However, other parent record types could be scoped to sites in the future. Feed elements that are always visible in all sites are filtered out. This value is valid only for the `UserProfile` feed.
- **QuestionsWithCandidateAnswers**—Feed elements that are questions that have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **QuestionsWithCandidateAnswersReviewedPublished**—Feed elements that are questions that have candidate answers that have been reviewed or published. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **Read**—Feed elements that are older than 30 days or are marked as read for the context user. Includes existing feed elements when the context user joined the group. This value is valid only for the `Record` feed of a group.
- **SolvedQuestions**—Feed elements that are questions and that have a best answer.
- **UnansweredQuestions**—Feed elements that are questions and that don’t have any answers.
• UnansweredQuestionsWithCandidateAnswers—Feed elements that are questions that don't have answers but have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
• Unread—Feed elements that are created in the past 30 days and aren't marked as read for the context user. This value is valid only for the Record feed of a group.
• UnsolvedQuestions—Feed elements that are questions and that don't have a best answer.

result
Type: ConnectApi.FeedElementPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
getFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, showInternalOnly, filter)


setTestGetFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, showInternalOnly, customFilter, result)

Register a ConnectApi.FeedElementPage object to be returned when getFeedElementsUpdatedSince is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

API Version
40.0

Signature
public static Void setTestGetFeedElementsUpdatedSince(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, Integer elementsPerBundle, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, String updatedSince, Boolean showInternalOnly, String customFilter, ConnectApi.FeedElementPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Value must be ConnectApi.FeedType.Record.
subjectId
Type: String
The ID of a case.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

elementsPerBundle
Type: Integer
Maximum number of feed elements per bundle. The default and maximum value is 10.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.

- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

customFilter
Type: String
Custom filter that applies only to the case feed. See customFeedFilter in the Metadata API Developer Guide for supported values.

result
Type: ConnectApi.FeedElementPage
Object containing test data.
Return Value
Type: Void

SEE ALSO:
getFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince, showInternalOnly, customFilter)


callTestGetRelatedPosts(communityId, feedElementId, filter, maxResults, result)

Register a ConnectApi.RelatedFeedPosts object to be returned when the matching
ConnectApi.getRelatedPosts(communityId, feedElementId, filter, maxResults) method is called
in a test context. Use the method with the same parameters or you receive an exception.

API Version
37.0

Signature
public static Void callTestGetRelatedPosts(String communityId, String feedElementId,
ConnectApi.RelatedFeedPostType filter, Integer maxResults, ConnectApi.RelatedFeedPosts
result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element. The feed element must be a question.

filter
Type: ConnectApi.RelatedFeedPostType
Specifies the type of related feed post. Values are:

• Answered—Related questions that have at least one answer.
• BestAnswer—Related questions that have a best answer.
• Generic—All types of related questions, including answered, with a best answer, and unanswered.
• Unanswered—Related questions that don’t have answers.

Generic is the default value.

maxResults
Type: Integer
The maximum number of results to return. You can return up to 25 results; 5 is the default.
result
Type: ConnectApi.RelatedFeedPosts
Object containing test data.
In version 37.0 and later, related feed posts are questions.

Return Value
Type: Void

**setTestGetTopUnansweredQuestions(communityId, result) (Pilot)**

Register a ConnectApi.FeedElementPage object to be returned when the matching ConnectApi.getTopUnansweredQuestions method is called in a test context. Use the method with the same parameters or you receive an exception.

Note: We provided top-five unanswered questions to selected customers through a pilot program that required agreement to specific terms and conditions. This pilot program is closed and not accepting new participants.

**API Version**

42.0

**Signature**

public static Void setTestGetTopUnansweredQuestions(String communityId, ConnectApi.FeedElementPage result)

**Parameters**

`communityId`
Type: String
ID of the Experience Cloud site.

`result`
Type: ConnectApi.FeedElementPage
Object containing test data.

**Return Value**

Type: Void

**SEE ALSO:**

getTopUnansweredQuestions(communityId) (Pilot)
**setTestGetTopUnansweredQuestions(communityId, filter, result) (Pilot)**

Register a `ConnectApi.FeedElementPage` object to be returned when the matching `ConnectApi.getTopUnansweredQuestions` method is called in a test context. Use the method with the same parameters or you receive an exception.

**Note:** We provided top-five unanswered questions to selected customers through a pilot program that required agreement to specific terms and conditions. This pilot program is closed and not accepting new participants.

**API Version**

42.0

**Signature**

```
public static Void setTestGetTopUnansweredQuestions(String communityId, 
ConnectApi.TopUnansweredQuestionsFilterType filter, ConnectApi.FeedElementPage result)
```

**Parameters**

- `communityId`
  - Type: String
  - ID of the Experience Cloud site.

- `filter`
  - Type: `ConnectApi.FeedFilter`
  - Specifies the filter for the feed. `UnansweredQuestionsWithCandidateAnswers` is the only valid value.

- `result`
  - Type: `ConnectApi.FeedElementPage`
  - Object containing test data.

**Return Value**

Type: Void

**SEE ALSO:**

- `getTopUnansweredQuestions(communityId, filter)` (Pilot)
- *Apex Developer Guide: Testing ConnectApi Code*

**setTestGetTopUnansweredQuestions(communityId, pageSize, result) (Pilot)**

Register a `ConnectApi.FeedElementPage` object to be returned when the matching `ConnectApi.getTopUnansweredQuestions` method is called in a test context. Use the method with the same parameters or you receive an exception.

**Note:** We provided top-five unanswered questions to selected customers through a pilot program that required agreement to specific terms and conditions. This pilot program is closed and not accepting new participants.
API Version
42.0

Signature
public static Void setTestGetTopUnansweredQuestions(String communityId, Integer pageSize, ConnectApi.FeedElementPage result)

Parameters

communityId
Type: String
ID of the Experience Cloud site.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 0 through 10. If you pass in null, the default size is 5.

result
Type: ConnectApi.FeedElementPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
getTopUnansweredQuestions(communityId, pageSize) (Pilot)

setTestGetTopUnansweredQuestions(communityId, filter, pageSize, result) (Pilot)
Register a ConnectApi.FeedElementPage object to be returned when the matching ConnectApi.getTopUnansweredQuestions method is called in a test context. Use the method with the same parameters or you receive an exception.

Note: We provided top-five unanswered questions to selected customers through a pilot program that required agreement to specific terms and conditions. This pilot program is closed and not accepting new participants.

API Version
42.0

Signature
public static Void setTestGetTopUnansweredQuestions(String communityId, ConnectApi.FeedFilter filter, Integer pageSize, ConnectApi.FeedElementPage result)
Parameters

`communityId`
Type: `String`
ID of the Experience Cloud site.

`filter`
Type: `ConnectApi.FeedFilter`
Specifies the filter for the feed. `UnansweredQuestionsWithCandidateAnswers` is the only valid value.

`pageSize`
Type: `Integer`
Specifies the number of items per page. Valid values are from 0 through 10. If you pass in `null`, the default size is 5.

`result`
Type: `ConnectApi.FeedElementPage`
Object containing test data.

Return Value
Type: Void

SEE ALSO:
- `getTopUnansweredQuestions(communityId, filter, pageSize)` (Pilot)

`setTestSearchFeedElements(communityId, q, result)`
Register a `ConnectApi.FeedElementPage` object to be returned when the matching `ConnectApi.searchFeedElements` method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
31.0

Signature

```java
public static Void setTestSearchFeedElements(String communityId, String q, ConnectApi.FeedElementPage result)
```

Parameters

`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

`q`
Type: `String`
Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

result
Type: ConnectApi.FeedElementPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
searchFeedElements(communityId, q)

setTestSearchFeedElements(communityId, q, sortParam, result)
Register a ConnectApi.FeedElementPage object to be returned when the matching ConnectApi.searchFeedElements method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
31.0

Signature
public static Void setTestSearchFeedElements(String communityId, String q, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedElementPage result)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
• CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
• CreatedDateDesc—Sorts by most recent creation date.
• LastModifiedDateDesc—Sorts by most recent activity.
MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.

Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

If you pass in `null`, the default value CreatedDateDesc is used.

**result**

Type: `ConnectApi.FeedElementPage`

Object containing test data.

**Return Value**

Type: Void

**SEE ALSO:**

`searchFeedElements(communityId, q, sortParam)`

*Apex Developer Guide: Testing ConnectApi Code*

### setTestSearchFeedElements(communityId, q, threadedCommentsCollapsed, result)

Register a `ConnectApi.FeedElementPage` object to be returned when the matching `ConnectApi.searchFeedElements` method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**

44.0

**Signature**

```java
public static Void setTestSearchFeedElements(String communityId, String q, Boolean threadedCommentsCollapsed, ConnectApi.FeedElementPage result)
```

**Parameters**

- `communityId`
  
  Type: `String`
  
  ID for an Experience Cloud site, internal, or `null`.

- `q`
  
  Type: `String`
  
  Required and can’t be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

- `threadedCommentsCollapsed`
  
  Type: `Boolean`
  
  Specifies whether to return threaded comments in a collapsed style (true) or not (false). If you pass in `null`, the default is `false`.

- `result`
  
  Type: `ConnectApi.FeedElementPage`
Object containing test data.

**Return Value**
Type: Void

SEE ALSO:
- `searchFeedElements(communityId, q, threadedCommentsCollapsed)`
- *Apex Developer Guide: Testing ConnectApi Code*

```java
public static Void setTestSearchFeedElements(String communityId, String q, String pageParam, Integer pageSize, ConnectApi.FeedElementPage result)
```

**Parameters**

- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.

- `q`
  Type: `String`
  Required and can't be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See *Wildcards*.

- `pageParam`
  Type: `String`
  Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

- `pageSize`
  Type: `Integer`
  Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

- `result`
  Type: `ConnectApi.FeedElementPage`
  Object containing test data.
Return Value
Type: Void

SEE ALSO:
searchFeedElements(communityId, q, pageParam, pageSize)

setTestSearchFeedElements(communityId, q, pageParam, pageSize, sortParam, result)

Register a ConnectApi.FeedElementPage object to be returned when the matching
ConnectApi.searchFeedElements method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
31.0

Signature
public static Void setTestSearchFeedElements(String communityId, String q, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedElementPage result)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
• CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
• CreatedDateDesc—Sorts by most recent creation date.
• LastModifiedDateDesc—Sorts by most recent activity.
• MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
• Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

If you pass in null, the default value CreatedDateDesc is used.

result
Type: ConnectApi.FeedElementPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
  searchFeedElements(communityId, q, pageParam, pageSize, sortParam)

setTestSearchFeedElements(communityId, q, pageParam, pageSize, threadedCommentsCollapsed, result)

Register a ConnectApi.FeedElementPage object to be returned when the matching ConnectApi.searchFeedElements method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
44.0

Signature
public static Void setTestSearchFeedElements(String communityId, String q, String pageParam, Integer pageSize, Boolean threadedCommentsCollapsed, ConnectApi.FeedElementPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

**pageSize**
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

**threadedCommentsCollapsed**
Type: Boolean
Specifies whether to return threaded comments in a collapsed style (true) or not (false). If you pass in null, the default is false.

**result**
Type: `ConnectApi.FeedElementPage`
Object containing test data.

**Return Value**
Type: Void

**SEE ALSO:**
searchFeedElements(communityId, q, pageParam, pageSize, threadedCommentsCollapsed)
*Apex Developer Guide: Testing ConnectApi Code*

```java
public static Void setTestSearchFeedElements(String communityId, String q, Integer recentCommentCount, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedElementPage result)
```

Register a `ConnectApi.FeedElementPage` object to be returned when the matching `ConnectApi.searchFeedElements` method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**
31.0

**Signature**

**Parameters**

**communityId**
Type: String
ID for an Experience Cloud site, internal, or null.

**q**
Type: String
Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
• CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
• CreatedDateDesc—Sorts by most recent creation date.
• LastModifiedDesc—Sorts by most recent activity.
• MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
• Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
If you pass in null, the default value CreatedDateDesc is used.

result
Type: ConnectApi.FeedElementPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
searchFeedElements(communityId, q, recentCommentCount, pageParam, pageSize, sortParam)

setTestSearchFeedElementsInFeed(communityId, feedType, q, result)
Register a ConnectApi.FeedElementPage object to be returned when the matching ConnectApi.searchFeedElementsInFeed method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
31.0
Signature

```java
public static Void setTestSearchFeedElementsInFeed(String communityId,
        ConnectApi.FeedType feedType, String q, ConnectApi.FeedElementPage result)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- `feedType`
  - Type: `ConnectApi.FeedType`
  - Type of feed. Valid values are `Company`, `DirectMessageModeration`, `Home`, `Moderation`, and `PendingReview`.

- `q`
  - Type: `String`
  - Required and can't be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See `Wildcards`.

- `result`
  - Type: `ConnectApi.FeedElementPage`
  - Object containing test data.

Return Value

Type: `Void`

SEE ALSO:

- `searchFeedElementsInFeed(communityId, feedType, q)`

```java
setTestSearchFeedElementsInFeed(communityId, feedType, pageParam, pageSize,
        sortParam, q, result)
```

Register a `ConnectApi.FeedElementPage` object to be returned when the matching
`ConnectApi.searchFeedElementsInFeed` method is called in a test context. Use the method with the same parameters
or you receive an exception.

API Version

31.0

Signature

```java
public static Void setTestSearchFeedElementsInFeed(String communityId,
        ConnectApi.FeedType feedType, String pageParam, Integer pageSize,
        ConnectApi.FeedSortOrder sortParam, String q, ConnectApi.FeedElementPage result)
```
Parameters

`communityId`
Type: String
ID for an Experience Cloud site, internal, or null.

`feedType`
Type: `ConnectApi.FeedType`
Type of feed. Valid values are Company, DirectMessageModeration, Home, Moderation, and PendingReview.

`pageParam`
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

`pageSize`
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

`sortParam`
Type: `ConnectApi.FeedSortOrder`
Values are:
- `CreatedDateAsc`—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- `CreatedDateDesc`—Sorts by most recent creation date.
- `LastModifiedDateDesc`—Sorts by most recent activity.
- `MostViewed`—Sorts by most viewed content. This sort order is available only for Home feeds when the `ConnectApi.FeedFilter` is UnansweredQuestions.
- `Relevance`—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

If you pass in null, the default value `CreatedDateDesc` is used.

`q`
Type: String
Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

`result`
Type: `ConnectApi.FeedElementPage`
Object containing test data.

Return Value
Type: Void

SEE ALSO:
s`earchFeedElementsInFeed(communityId, feedType, pageParam, pageSize, sortParam, q)`

*Apex Developer Guide: Testing ConnectApi Code*
`setTestSearchFeedElementsInFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, q, result)`

Register a `ConnectApi.FeedElementPage` object to be returned when the matching `ConnectApi.searchFeedElementsInFeed` method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**

31.0

**Signature**

```java
public static Void setTestSearchFeedElementsInFeed(String communityId, ConnectApi.FeedType feedType, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q, ConnectApi.FeedElementPage result)
```

**Parameters**

- `communityId`
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- `feedType`
  - Type: `ConnectApi.FeedType`
  - Type of feed. Valid values are Company, DirectMessageModeration, Home, Moderation, and PendingReview.

- `recentCommentCount`
  - Type: Integer
  - Maximum number of comments to return with each feed element. The default value is 3.

- `density`
  - Type: `ConnectApi.FeedDensity`
  - Specify the amount of content in a feed.
    - AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
    - FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

- `pageParam`
  - Type: String
  - Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

- `pageSize`
  - Type: Integer
  - Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

- `sortParam`
  - Type: `ConnectApi.FeedSortOrder`
Values are:

- **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- **Relevance**—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.

If you pass in `null`, the default value `CreatedDateDesc` is used.

`q`

Type: **String**

Required and can't be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See [Wildcards](#).

`result`

Type: **ConnectApi.FeedElementPage**

Object containing test data.

**Return Value**

Type: Void

**SEE ALSO:**

- `searchFeedElementsInFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, q)`
- [Apex Developer Guide: Testing ConnectApi Code](#)

**setTestSearchFeedElementsInFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, q, filter, result)**

Register a `ConnectApi.FeedElementPage` object to be returned when the matching `ConnectApi.searchFeedElementsInFeed` method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**

32.0

**Signature**

```java
public static Void setTestSearchFeedElementsInFeed(String communityId,
ConnectApi.FeedType feedType, Integer recentCommentCount, ConnectApi.FeedDensity density,
String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q,
ConnectApi.FeedFilter filter, ConnectApi.FeedElementPage result)
```
Parameters

`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

`feedType`
Type: `ConnectApi.FeedType`
The type of feed. The only valid value is `Home`.

`recentCommentCount`
Type: `Integer`
Maximum number of comments to return with each feed element. The default value is 3.

`density`
Type: `ConnectApi.FeedDensity`
Specify the amount of content in a feed.
- `AllUpdates`—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- `FewerUpdates`—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

`pageParam`
Type: `String`
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

`pageSize`
Type: `Integer`
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

`sortParam`
Type: `ConnectApi.FeedSortOrder`
Values are:
- `CreateDateAsc`—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- `CreateDateDesc`—Sorts by most recent creation date.
- `LastModifiedDateDesc`—Sorts by most recent activity.
- `MostViewed`—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- `Relevance`—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.
If you pass in `null`, the default value `CreateDateDesc` is used.

`q`
Type: `String`
Required and can't be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

`filter`
Type: `ConnectApi.FeedFilter`
Specifies the feed filters.

- **AllQuestions**—Feed elements that are questions.
- **AuthoredBy**—Feed elements authored by the user profile owner. This value is valid only for the UserProfile feed.
- **CommunityScoped**—Feed elements that are scoped to Experience Cloud sites. Currently, these feed elements have a User or a Group parent record. However, other parent record types could be scoped to sites in the future. Feed elements that are always visible in all sites are filtered out. This value is valid only for the UserProfile feed.
- **QuestionsWithCandidateAnswers**—Feed elements that are questions that have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **QuestionsWithCandidateAnswersReviewedPublished**—Feed elements that are questions that have candidate answers that have been reviewed or published. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **Read**—Feed elements that are older than 30 days or are marked as read for the context user. Includes existing feed elements when the context user joined the group. This value is valid only for the Record feed of a group.
- **SolvedQuestions**—Feed elements that are questions and that have a best answer.
- **UnansweredQuestions**—Feed elements that are questions and that don’t have any answers.
- **UnansweredQuestionsWithCandidateAnswers**—Feed elements that are questions that don’t have answers but have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **Unread**—Feed elements that are created in the past 30 days and aren’t marked as read for the context user. This value is valid only for the Record feed of a group.
- **UnsolvedQuestions**—Feed elements that are questions and that don’t have a best answer.

```result
Type: ConnectApi.FeedElementPage
Object containing test data.
```

**Return Value**

Type: Void

SEE ALSO:

- `searchFeedElementsInFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, q, filter)`

```setTestSearchFeedElementsInFeed(communityId, feedType, subjectId, q, result)`

Register a ConnectApi.FeedElementPage object to be returned when the matching ConnectApi.searchFeedElementsInFeed method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**

31.0
Signature

```java
public static Void setTestSearchFeedElementsInFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, String q, ConnectApi.FeedElementPage result)
```

Parameters

- `communityId`
  - Type: `String`  
  - ID for an Experience Cloud site, `internal`, or `null`.

- `feedType`
  - Type: `ConnectApi.FeedType`  
  - Type of feed. Valid values include every `ConnectApi.FeedType` except `Company`, `DirectMessages`, `Filter`, `Landing`, `Streams`, and `Topics`.

- `subjectId`
  - Type: `String`  
  - If `feedType` is `Record`, `subjectId` can be any record ID, including a group ID. If `feedType` is `UserProfile`, `subjectId` can be any user ID. If `feedType` is any other value, `subjectId` must be the ID of the context user or the alias `me`.

- `q`
  - Type: `String`  
  - Required and can't be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See `Wildcards`.

- `result`
  - Type: `ConnectApi.FeedElementPage`  
  - Object containing test data.

Return Value

- Type: `Void`

SEE ALSO:

- `searchFeedElementsInFeed(communityId, feedType, subjectId, q)`  

```
setTestSearchFeedElementsInFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam, q, result)
```

Register a `ConnectApi.FeedElementPage` object to be returned when the matching `ConnectApi.searchFeedElementsInFeed` method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version

- `31.0`
public static Void setTestSearchFeedElementsInFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q, ConnectApi.FeedElementPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Type of feed. Valid values include every ConnectApi.FeedType except Company, DirectMessages, Filter, Landing, Streams, and Topics.

subjectId
Type: String
If feedType is Record, subjectId can be any record ID, including a group ID. If feedType is UserProfile, subjectId can be any user ID. If the feedType is any other value, subjectId must be the ID of the context user or the alias me.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Order of feed items in the feed.

- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
If you pass in null, the default value CreatedDateDesc is used.

q
Type: String
Search term. Searches keywords in the user or group name. A minimum of one character is required. This parameter doesn’t support wildcards. This parameter is required.
result
   Type: ConnectApi.FeedElementPage
   Object containing test data.

Return Value
Type: Void

SEE ALSO:
  searchFeedElementsInFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam, q)

setTestSearchFeedElementsInFeed(communityId, feedType, subjectId,
recentCommentCount, density, pageParam, pageSize, sortParam, q, result)

Register a ConnectApi.FeedElementPage object to be returned when the matching
ConnectApi.searchFeedElementsInFeed method is called in a test context. Use the method with the same parameters
or you receive an exception.

API Version
31.0

Signature
public static Void setTestSearchFeedElementsInFeed(String communityId,
ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount,
ConnectApi.FeedDensity density, String pageParam, Integer pageSize,
ConnectApi.FeedSortOrder sortParam, String q, ConnectApi.FeedElementPage result)

Parameters
communityId
   Type: String
   ID for an Experience Cloud site, internal, or null.

feedType
   Type: ConnectApi.FeedType
   Type of feed. Valid values include every ConnectApi.FeedType except Company, DirectMessages, Filter,
   Landing, Streams, and Topics.

subjectId
   Type: String
   If feedType is Record, subjectId can be any record ID, including a group ID. If feedType is UserProfile,
   subjectId can be any user ID. If the feedType is any other value, subjectId must be the ID of the context user or the
   alias me.

recentCommentCount
   Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

density
Type: `ConnectApi.FeedDensity`
Specify the amount of content in a feed.

- **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: `String`
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

pageSize
Type: `Integer`
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

sortParam
Type: `ConnectApi.FeedSortOrder`
Values are:

- **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for Home feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- **Relevance**—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

If you pass in `null`, the default value `CreatedDateDesc` is used.

q
Type: `String`
Required and can’t be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See [Wildcards](https://developer.salesforce.com/docs/atlas.en-us.apexcode.meta/apexcode/sldemo apex developer guide.htm).

result
Type: `ConnectApi.FeedElementPage`
Object containing test data.

Return Value
Type: Void

SEE ALSO:

- `searchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q)`

APEX Developer Guide: Testing ConnectApi Code
Register a ConnectApi.FeedElementPage object to be returned when searchFeedElementsInFeed is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

**API Version**

35.0

**Signature**

```java
public static Void setTestSearchFeedElementsInFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q, ConnectApi.FeedFilter filter, ConnectApi.FeedElementPage result)
```

**Parameters**

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- **feedType**
  - Type: ConnectApi.FeedType
  - Value must be ConnectApi.FeedType.UserProfile.

- **subjectId**
  - Type: String
  - ID of any user. To specify the context user, use the user ID or the alias me.

- **recentCommentCount**
  - Type: Integer
  - Maximum number of comments to return with each feed element. The default value is 3.

- **density**
  - Type: ConnectApi.FeedDensity
  - The amount of content in a feed.
  - **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
  - **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

- **pageParam**
  - Type: String
  - Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in null, the first page is returned.
pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
• CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
• CreatedDateDesc—Sorts by most recent creation date.
• LastModifiedDateDesc—Sorts by most recent activity.
• MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
• Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
If you pass in null, the default value CreatedDateDesc is used.

q
Type: String
One or more keywords to search for in the feed elements visible to the context user. The search string can contain wildcards and must contain at least two characters that aren’t wildcards. See Wildcards.

filter
Type: ConnectApi.FeedFilter
Value must be ConnectApi.FeedFilter.CommunityScoped. Filters the feed to include only feed elements that are scoped to Experience Cloud sites. Feed elements that are always visible in all sites are filtered out. Currently, feed elements scoped to sites have a User or a Group parent record. However, other parent record types could be scoped to sites in the future.

result
Type: ConnectApi.FeedElementPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
searchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, filter)

setTestSearchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, customFilter, result)
Register a ConnectApi.FeedElementPage object to be returned when the matching ConnectApi.searchFeedElementsInFeed method is called in a test context. Use the method with the same parameters or you receive an exception.
API Version

40.0

Signature

public static Void setTestSearchFeedElementsInFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q, String customFilter, ConnectApi.FeedElementPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Value must be ConnectApi.FeedType.Record.

subjectId
Type: String
The ID of a case.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

density
Type: ConnectApi.FeedDensity
The amount of content in a feed.

• AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
• FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
• **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.

• **CreatedDateDesc**—Sorts by most recent creation date.

• **LastModifiedDateDesc**—Sorts by most recent activity.

• **MostViewed**—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.

• **Relevance**—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

If you pass in null, the default value CreatedDateDesc is used.

**q**

Type: String

One or more keywords to search for in the feed elements visible to the context user. The search string can contain wildcards and must contain at least two characters that aren’t wildcards. See Wildcards.

**customFilter**

Type: String

Custom filter that applies only to the case feed. See customFeedFilter in the Metadata API Developer Guide for supported values.

**result**

Type: ConnectApi.FeedElementPage

Object containing test data.

**Return Value**

Type: Void

SEE ALSO:

searchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, customFilter)


```java
setTestSearchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q,
    showInternalOnly, result)
```

Register a ConnectApi.FeedElementPage object to be returned when the matching ConnectApi.searchFeedElementsInFeed method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**

31.0

**Signature**

```java
public static Void setTestSearchFeedElementsInFeed(String communityId,
    ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount,
    ConnectApi.FeedDensity density, String pageParam, Integer pageSize,
```
ConnectApi.FeedSortOrder sortParam, String q, Boolean showInternalOnly, ConnectApi.FeedElementPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Value must be ConnectApi.FeedType.Record.

subjectId
Type: String
Any record ID, including a group ID.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed element. The default value is 3.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
If you pass in null, the default value CreatedDateDesc is used.
q
  Type: String
  Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

showInternalOnly
  Type: Boolean
  Specifies whether to show only feed elements from internal (non-Experience Cloud site) users (true), or not (false). The default value is false.

result
  Type: ConnectApi.FeedElementPage
  Object containing test data.

Return Value
  Type: Void

SEE ALSO:
  searchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, showInternalOnly)

setTestSearchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, showInternalOnly, filter, result)

Register a ConnectApi.FeedElementPage object to be returned when the matching ConnectApi.searchFeedElementsInFeed method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
  32.0

Signature
  public static Void setTestSearchFeedElementsInFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q, Boolean showInternalOnly, ConnectApi.FeedFilter filter, ConnectApi.FeedElementPage result)

Parameters
  communityId
    Type: String
    ID for an Experience Cloud site, internal, or null.
**feedType**
Type: `ConnectApi.FeedType`
Value must be `ConnectApi.FeedType.Record`.

**subjectId**
Type: `String`
Any record ID, including a group ID.

**recentCommentCount**
Type: `Integer`
Maximum number of comments to return with each feed element. The default value is 3.

**density**
Type: `ConnectApi.FeedDensity`
Specify the amount of content in a feed.
- **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**
Type: `String`
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**
Type: `Integer`
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**
Type: `ConnectApi.FeedSortOrder`
Values are:
- **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- **Relevance**—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.

If you pass in `null`, the default value `CreatedDateDesc` is used.

**q**
Type: `String`
Required and can’t be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

**showInternalOnly**
Type: `Boolean`
Specifies whether to show only feed elements from internal (non-Experience Cloud site) users (true), or not (false). The default value is false.

**filter**

Type: `ConnectApi.FeedFilter`

Specifies the feed filters.

- **AllQuestions**—Feed elements that are questions.
- **AuthoredBy**—Feed elements authored by the user profile owner. This value is valid only for the `UserProfile` feed.
- **CommunityScoped**—Feed elements that are scoped to Experience Cloud sites. Currently, these feed elements have a User or a Group parent record. However, other parent record types could be scoped to sites in the future. Feed elements that are always visible in all sites are filtered out. This value is valid only for the `UserProfile` feed.
- **QuestionsWithCandidateAnswers**—Feed elements that are questions that have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **QuestionsWithCandidateAnswersReviewedPublished**—Feed elements that are questions that have candidate answers that have been reviewed or published. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **Read**—Feed elements that are older than 30 days or are marked as read for the context user. Includes existing feed elements when the context user joined the group. This value is valid only for the `Record` feed of a group.
- **SolvedQuestions**—Feed elements that are questions and that have a best answer.
- **UnansweredQuestions**—Feed elements that are questions and that don’t have any answers.
- **UnansweredQuestionsWithCandidateAnswers**—Feed elements that are questions that don’t have answers but have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.
- **Unread**—Feed elements that are created in the past 30 days and aren’t marked as read for the context user. This value is valid only for the `Record` feed of a group.
- **UnsolvedQuestions**—Feed elements that are questions and that don’t have a best answer.

**result**

Type: `ConnectApi.FeedElementPage`

Object containing test data.

**Return Value**

Type: Void

SEE ALSO:

- `searchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, showInternalOnly, filter)`
- *Apex Developer Guide: Testing ConnectApi Code*
setTestSearchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, showInternalOnly, customFilter, result)

Register a ConnectApi.FeedElementPage object to be returned when the matching ConnectApi.searchFeedElementsInFeed method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
40.0

Signature
public static Void setTestSearchFeedElementsInFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q, Boolean showInternalOnly, String customFilter, ConnectApi.FeedElementPage result)

Parameters

communityId
Type: String

ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType

Value must be ConnectApi.FeedType.Record.

subjectId
Type: String

The ID of a case.

recentCommentCount
Type: Integer

Maximum number of comments to return with each feed element. The default value is 3.

density
Type: ConnectApi.FeedDensity

Specify the amount of content in a feed.

- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String

Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.
pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
• CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
• CreatedDateDesc—Sorts by most recent creation date.
• LastModifiedDateDesc—Sorts by most recent activity.
• MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
• Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds. If you pass in null, the default value CreatedDateDesc is used.

q
Type: String
Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

showInternalOnly
Type: Boolean
Specifies whether to show only feed elements from internal (non-Experience Cloud site) users (true), or not (false). The default value is false.

customFilter
Type: String
Custom filter that applies only to the case feed. See customFeedFilter in the Metadata API Developer Guide for supported values.

result
Type: ConnectApi.FeedElementPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
searchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, showInternalOnly, customFilter)
setTestSearchFeedElementsInFilterFeed(communityId, subjectId, keyPrefix, q, result)

Register a ConnectApi.FeedElementPage object to be returned when the matching
ConnectApi.searchFeedElementsInFilterFeed method is called in a test context. Use the method with the same
parameters or you receive an exception.

API Version
31.0

Signature
public static Void setTestSearchFeedElementsInFilterFeed(String communityId, String
subjectId, String keyPrefix, String q, ConnectApi.FeedElementPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subjectId
Type: String
ID of the context user or the alias me.

keyPrefix
Type: String
A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

q
Type: String
Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

result
Type: ConnectApi.FeedElementPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:

searchFeedElementsInFilterFeed(communityId, subjectId, keyPrefix, q)

setTestSearchFeedElementsInFilterFeed(communityId, subjectId, keyPrefix, pageParam, pageSize, sortParam, q, result)

Register a ConnectApi.FeedElementPage object to be returned when the matching ConnectApi.searchFeedElementsInFilterFeed method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
31.0

Signature
public static Void setTestSearchFeedElementsInFilterFeed(String communityId, String subjectId, String keyPrefix, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q, ConnectApi.FeedElementPage result)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subjectId
Type: String
ID of the context user or the alias me.

keyPrefix
Type: String
A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds. If you pass in null, the default value CreatedDateDesc is used.

q
Type: String
Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

result
Type: ConnectApi.FeedElementPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
searchFeedElementsInFilterFeed(communityId, subjectId, keyPrefix, pageParam, pageSize, sortParam, q)

setTestSearchFeedElementsInFilterFeed(communityId, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, sortParam, q, result)
Register a ConnectApi.FeedElementPage object to be returned when the matching ConnectApi.searchFeedElementsInFilterFeed method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
31.0

Signature
public static Void setTestSearchFeedElementsInFilterFeed(String communityId, String subjectId, String keyPrefix, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q, ConnectApi.FeedElementPage result)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subjectId
Type: String
ID of the context user or the alias me.

keyPrefix
Type: String
A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

**recentCommentCount**
- **Type:** Integer
- Maximum number of comments to return with each feed element. The default value is 3.

**density**
- **Type:** `ConnectApi.FeedDensity`
- Specify the amount of content in a feed.
  - **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
  - **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**
- **Type:** String
- Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**
- **Type:** Integer
- Specifies the number of feed elements per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**
- **Type:** `ConnectApi.FeedSortOrder`
- Values are:
  - **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
  - **CreatedDateDesc**—Sorts by most recent creation date.
  - **LastModifiedDesc**—Sorts by most recent activity.
  - **MostViewed**—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
  - **Relevance**—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.
- If you pass in `null`, the default value `CreatedDateDesc` is used.

**q**
- **Type:** String
- Required and can’t be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

**result**
- **Type:** `ConnectApi.FeedElementPage`
- Object containing test data.
Return Value
Type: Void

SEE ALSO:
searchFeedElementsInFilterFeed(communityId, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, sortParam, q)

setTestSearchStreams(communityId, q, result)
Register a ConnectApi.ChatterStreamPage object to be returned when the matching ConnectApi.searchStream(communityId, q) method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
40.0

Signature
public static Void setTestSearchStreams(String communityId, String q, ConnectApi.ChatterStreamPage result)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

result
Type: ConnectApi.ChatterStreamPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
searchStreams(communityId, q)
setTestSearchStreams(communityId, q, sortParam, result)

Register a `ConnectApi.ChatterStreamPage` object to be returned when the matching `ConnectApi.searchStream(communityId, q, sortParam)` method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version

40.0

Signature

```java
public static Void setTestSearchStreams(String communityId, String q,
ConnectApi.SortOrder sortParam, ConnectApi.ChatterStreamPage result)
```

Parameters

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **q**
  - Type: `String`
  - Required and can’t be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See `Wildcards`.

- **sortParam**
  - Type: `ConnectApi.SortOrder`
  - Specifies the sort order. Values are:
    - `Ascending`—Items are in ascending alphabetical order (A-Z).
    - `Descending`—Items are in descending alphabetical order (Z-A).
    - `MostRecentlyViewed`—Items are in descending chronological order by view. This sort order is valid only for Chatter feed streams.
  - If not specified, default value is `Ascending`.

- **result**
  - Type: `ConnectApi.ChatterStreamPage`
  - Object containing test data.

Return Value

- Type: `Void`

SEE ALSO:

- `searchStreams(communityId, q, sortParam)`
- *Apex Developer Guide: Testing ConnectApi Code*
**setTestSearchStreams**(communityId, q, pageParam, pageSize, result)

Register a `ConnectApi.ChatterStreamPage` object to be returned when the matching `ConnectApi.searchStreams(communityId, q, pageParam, pageSize)` method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**

40.0

**Signature**

```java
public static Void setTestSearchStreams(String communityId, String q, Integer pageParam, Integer pageSize, ConnectApi.ChatterStreamPage result)
```

**Parameters**

`communityId`

Type: `String`

ID for an Experience Cloud site, `internal`, or `null`.

`q`

Type: `String`

Required and can't be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See [Wildcards](#).

`pageParam`

Type: `Integer`

Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

`pageSize`

Type: `Integer`

Specifies the number of items per page. Valid values are from 1 to 250. The default size is 25.

`result`

Type: `ConnectApi.ChatterStreamPage`

Object containing test data.

**Return Value**

Type: `Void`

**SEE ALSO:**

- `searchStreams(communityId, q, pageParam, pageSize)`
- *Apex Developer Guide: Testing ConnectApi Code*
setTestSearchStreams(communityId, q, pageParam, pageSize, sortParam, result)

Register a ConnectApi.ChatterStreamPage object to be returned when the matching
ConnectApi.searchStreams(communityId, q, pageParam, pageSize, sortParam) method is called in
a test context. Use the method with the same parameters or you receive an exception.

API Version
40.0

Signature
public static Void setTestSearchStreams(String communityId, String q, Integer pageParam,
Integer pageSize, ConnectApi.SortOrder sortParam, ConnectApi.ChatterStreamPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See wildcards.

pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 to 250. The default size is 25.

sortParam
Type: ConnectApi.SortOrder
Specifies the sort order. Values are:
• Ascending—Items are in ascending alphabetical order (A-Z).
• Descending—Items are in descending alphabetical order (Z-A).
• MostRecentlyViewed—Items are in descending chronological order by view. This sort order is valid only for Chatter feed streams.

If not specified, default value is Ascending.

result
Type: ConnectApi.ChatterStreamPage
Object containing test data.
Return Value
Type: Void

SEE ALSO:
searchStreams(communityId, q, pageParam, pageSize, sortParam)

setTestSearchStreams(communityId, q, pageParam, pageSize, sortParam, globalScope, result)

Register a ConnectApi.ChatterStreamPage object to be returned when the matching
ConnectApi.searchStreams(communityId, q, pageParam, pageSize, sortParam, globalScope)
method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
41.0

Signature
public static Void setTestSearchStreams(String communityId, String q, Integer pageParam, Integer pageSize, ConnectApi.SortOrder sortParam, Boolean globalScope, ConnectApi.ChatterStreamPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 to 250. The default size is 25.

sortParam
Type: ConnectApi.SortOrder
Specifies the sort order. Values are:
- Ascending—Items are in ascending alphabetical order (A-Z).
- Descending—Items are in descending alphabetical order (Z-A).
• MostRecentlyViewed—Items are in descending chronological order by view. This sort order is valid only for Chatter feed streams.

If not specified, default value is Ascending.

`globalScope`
  Type: Boolean
  Specifies whether to get streams from all the context user’s Experience Cloud sites, regardless of the `communityId` value.

`result`
  Type: `ConnectApi.ChatterStreamPage`
  Object containing test data.

**Return Value**

Type: Void

**Retired ChatterFeeds Methods**

The following methods for ChatterFeeds are retired.

IN THIS SECTION:

- `deleteFeedItem(communityId, feedItemId)`
  Delete a feed item.
- `getCommentsForFeedItem(communityId, feedItemId)`
  Get comments for a feed item.
- `getCommentsForFeedItem(communityId, feedItemId, pageParam, pageSize)`
  Get a page of comments for a feed item.
- `getFeedItem(communityId, feedItemId)`
  Get a feed item.
- `getFeedItemBatch(communityId, feedItemIds)`
  Get a list of feed items.
- `getFeedItemsFromFeed(communityId, feedType)`
  Get feed items from the Company, Home, and Moderation feeds.
- `getFeedItemsFromFeed(communityId, feedType, pageParam, pageSize, sortParam)`
  Get a page of sorted feed items from the Company, Home, and Moderation feeds.
- `getFeedItemsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam)`
  Get a page of sorted feed items from the Company, Home, and Moderation feeds. Each feed item contains no more than the specified number of comments.
- `getFeedItemsFromFeed(communityId, feedType, subjectId)`
  Get feed items from a feed for a user or record.
- `getFeedItemsFromFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam)`
  Get a page of sorted feed items from a feed for a user or record.
getFeedItemsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam)
Get a page of sorted feed items from a feed for a user or record. Each feed item includes no more than the specified number of comments.

getFeedItemsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, showInternalOnly)
Get a page of sorted feed items from a record feed for a user or record. Each feed item includes no more than the specified number of comments. Specify whether to return feed items posted by internal (non-Experience Cloud site) users only.

getFeedItemsFromFilterFeed(communityId, subjectId, keyPrefix)
Get feed items from a feed filtered by a key prefix for a user.

getFeedItemsFromFilterFeed(communityId, subjectId, keyPrefix, pageParam, pageSize, sortParam)
Get a page of sorted feed items from a feed filtered by a key prefix for a user.

getFeedItemsFromFilterFeed(communityId, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, sortParam)
Get a page of sorted feed items from a feed filtered by a key prefix for a user. Each feed item contains no more than the specified number of comments.

getFeedItemsFromFilterFeedUpdatedSince(communityId, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, updatedSince)
Get a page of feed items from a feed filtered by a key prefix for a user. Include only feed items that have been updated since the time specified in the updatedSince parameter.

getFeedItemsUpdatedSince(communityId, feedType, recentCommentCount, density, pageParam, pageSize, updatedSince)
Get a page of feed items from the Company, Home, and Moderation feeds. Include only feed items that have been updated since the time specified in the updatedSince parameter. Each feed item contains no more than the specified number of comments.

getFeedItemsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince)
Get a page of feed items from the Files, Groups, News, People, and Record feeds. Include only feed items that have been updated since the time specified in the updatedSince parameter. Each feed item contains no more than the specified number of comments.

getFeedItemsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince, showInternalOnly)
Get a page of feed items from a record feed. Include only feed items that have been updated since the time specified in the updatedSince parameter. Specify whether to return feed items posted by internal (non-Experience Cloud site) users only.

getFeedPoll(communityId, feedItemId)
Get the poll associated with a feed item.

getLikesForFeedItem(communityId, feedItemId)
Get likes for a feed item.

getLikesForFeedItem(communityId, feedItemId, pageParam, pageSize)
Get a page of likes for a feed item.

likeFeedItem(communityId, feedItemId)
Like a feed item for the context user.

postComment(communityId, feedItemId, text)
Post a plain-text comment to a feed item.

postComment(communityId, feedItemId, comment, feedItemFileUpload)
Post a rich-text comment to a feed item. Use this method to include mentions and to attach a file to a comment.
postFeedElement(communityId, feedElement, feedElementFileUpload)
Post a rich-text feed element. Include mentions and hashtag topics, attach a file to a feed element, and associate action link groups with a feed element. You can also use this method to share a feed element and add a comment.

postFeedItem(communityId, feedType, subjectId, text)
Post a plain-text feed item.

postFeedItem(communityId, feedType, subjectId, feedItemInput, feedItemFileUpload)
Post a rich-text feed item to a feed. Use this method to include mentions and hashtag topics and to attach a file to a feed item. You can also use this method to share a feed item and add a comment.

searchFeedItems(communityId, q)
Get the feed items that match the search criteria.

searchFeedItems(communityId, q, sortParam)
Get the sorted feed items that match the search criteria.

searchFeedItems(communityId, q, pageParam, pageSize)
Get a page of feed items that match the search criteria.

searchFeedItems(communityId, q, pageParam, pageSize, sortParam)
Get a page of sorted feed items that match the search criteria.

searchFeedItems(communityId, q, recentCommentCount, pageParam, pageSize, sortParam)
Get a page of sorted feed items that match the search criteria. Each feed item includes no more than the specified number of comments.

searchFeedItemsInFeed(communityId, feedType, q)
Get the feed items from the Company, Home, and Moderation feeds that match the search criteria.

searchFeedItemsInFeed(communityId, feedType, pageParam, pageSize, sortParam, q)
Get a page of sorted feed items from the Company, Home, and Moderation feeds that match the search criteria.

searchFeedItemsInFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, q)
Get a page of sorted feed items from the Company, Home, and Moderation feeds that match the search criteria. Each feed item includes no more than the specified number of comments.

searchFeedItemsInFeed(communityId, feedType, subjectId, q)
Get the feed items from a feed that match the search criteria.

searchFeedItemsInFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam, q)
Get a page of sorted feed items from a feed for a user or record that match the search criteria.

searchFeedItemsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q)
Get a page of sorted feed items from a feed that match the search criteria. Each feed item includes no more than the specified number of comments.

searchFeedItemsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, showInternalOnly)
Get a page of sorted feed items from a feed for a user or record that match the search criteria. Each feed item includes no more than the specified number of comments. Specify whether to return feed items posted by internal (non-Experience Cloud site) users only.

searchFeedItemsInFilterFeed(communityId, subjectId, keyPrefix, q)
Get the feed items that match the search criteria from a feed filtered by a key prefix for a user.

searchFeedItemsInFilterFeed(communityId, subjectId, keyPrefix, pageParam, pageSize, sortParam, q)
Get a page of sorted feed items that match the search criteria from a feed filtered by a key prefix for a user.
searchFeedItemsInFilterFeed(communityId, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, sortParam, q)

Get a page of sorted feed items that match the search criteria from a feed filtered by a key prefix for a user. Each feed item includes no more than the specified number of comments.

shareFeedElement(communityId, subjectId, feedElementType, originalFeedElementId)

Share the originalFeedElementId as the context user.

shareFeedItem(communityId, feedType, subjectId, originalFeedItemId)

Share the originalFeedItemId to the feed specified by the feedType.

updateBookmark(communityId, feedId, isBookmarkedByCurrentUser)

Bookmark a feed item or remove a bookmark from a feed item.

voteOnFeedPoll(communityId, feedItemId, myChoiceId)

Vote or change your vote on a feed poll.

setTestGetFeedItemsFromFeed(communityId, feedType, result)

Register a ConnectApi.FeedItemPage object to be returned when getFeedItemsFromFeed is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

setTestGetFeedItemsFromFeed(communityId, feedType, pageParam, pageSize, sortParam, result)

Register a ConnectApi.FeedItemPage object to be returned when getFeedItemsFromFeed is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

setTestGetFeedItemsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, result)

Register a ConnectApi.FeedItemPage object to be returned when getFeedItemsFromFeed is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

setTestGetFeedItemsFromFeed(communityId, feedType, subjectId, result)

Register a ConnectApi.FeedItemPage object to be returned when getFeedItemsFromFeed is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

setTestGetFeedItemsFromFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam, result)

Register a ConnectApi.FeedItemPage object to be returned when getFeedItemsFromFeed is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

setTestGetFeedItemsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, result)

Register a ConnectApi.FeedItemPage object to be returned when getFeedItemsFromFeed is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

setTestGetFeedItemsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, showInternalOnly, result)

Register a ConnectApi.FeedItemPage object to be returned when getFeedItemsFromFeed is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

setTestGetFeedItemsFromFilterFeed(communityId, subjectId, keyPrefix, result)

Register a ConnectApi.FeedItemPage object to be returned when the matching getFeedItemsFromFeed method is called in a test context. Use the method with the same parameters or the code throws an exception.

setTestGetFeedItemsFromFilterFeed(communityId, subjectId, keyPrefix, pageParam, pageSize, sortParam, result)

Register a ConnectApi.FeedItemPage object to be returned when the matching getFeedItemsFromFilterFeed method is called in a test context. Use the method with the same parameters or the code throws an exception.
setTestGetFeedItemsFromFilterFeed

Register a ConnectApi.FeedItemPage object to be returned when the matching getFeedItemsFromFilterFeed method is called in a test context. Use the method with the same parameters or the code throws an exception.

setTestGetFeedItemsFromFilterFeedUpdatedSince

Register a ConnectApi.FeedItemPage object to be returned when the getFeedItemsFromFilterFeedUpdatedSince method is called in a test context.

setTestGetFeedItemsUpdatedSince

Register a ConnectApi.FeedItemPage object to be returned when getFeedItemsUpdatedSince is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

setTestGetFeedItemsUpdatedSince

Register a ConnectApi.FeedItemPage object to be returned when the matching ConnectApi.searchFeedItemsInFeed method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchFeedItemsInFeed

Register a ConnectApi.FeedItemPage object to be returned when the matching ConnectApi.searchFeedItemsInFeed method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchFeedItems

Register a test feed item page to be returned when searchFeedItems (communityId, q) is called during a test.

setTestSearchFeedItems

Register a test feed item page to be returned when searchFeedItems (String, String, ConnectApi.FeedSortOrder) is called during a test.

setTestSearchFeedItems

Register a test feed item page to be returned when searchFeedItems (String, String, String, Integer) is called during a test.

setTestSearchFeedItems

Register a test feed item page to be returned when searchFeedItems (String, String, String, Integer, ConnectApi.FeedSortOrder) is called during a test.

setTestSearchFeedItems

Register a test feed item page to be returned when searchFeedItems (communityId, q, recentCommentCount, pageParam, pageSize, sortParam, result) is called during a test.
setTestSearchFeedItemsInFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, q, result)

Register a `ConnectApi.FeedItemPage` object to be returned when the matching `ConnectApi.searchFeedItemsInFeed` method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchFeedItemsInFeed(communityId, feedType, subjectId, q, result)

Register a `ConnectApi.FeedItemPage` object to be returned when the matching `ConnectApi.searchFeedItemsInFeed` method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchFeedItemsInFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam, q, result)

Register a `ConnectApi.FeedItemPage` object to be returned when the matching `ConnectApi.searchFeedItemsInFeed` method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchFeedItemsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, result)

Register a `ConnectApi.FeedItemPage` object to be returned when the matching `ConnectApi.searchFeedItemsInFeed` method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchFeedItemsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, result)

Register a `ConnectApi.FeedItemPage` object to be returned when the matching `ConnectApi.searchFeedItemsInFeed` method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchFeedItemsInFilterFeed(communityId, subjectId, keyPrefix, q, result)

Register a `ConnectApi.FeedItemPage` object to be returned when the matching `ConnectApi.searchFeedItemsInFilterFeed` method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchFeedItemsInFilterFeed(communityId, feedType, subjectId, keyPrefix, pageParam, pageSize, sortParam, q, result)

Register a `ConnectApi.FeedItemPage` object to be returned when the matching `ConnectApi.searchFeedItemsInFilterFeed` method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchFeedItemsInFilterFeed(communityId, feedType, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, sortParam, q, result)

Register a `ConnectApi.FeedItemPage` object to be returned when the matching `ConnectApi.searchFeedItemsInFilterFeed` method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchFeedItemsInFilterFeed(communityId, feedType, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, sortParam, q, result)

Register a `ConnectApi.FeedItemPage` object to be returned when the matching `ConnectApi.searchFeedItemsInFilterFeed` method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchFeedItemsInFilterFeed(communityId, feedType, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, sortParam, q, result)

Register a `ConnectApi.FeedItemPage` object to be returned when the matching `ConnectApi.searchFeedItemsInFilterFeed` method is called in a test context. Use the method with the same parameters or you receive an exception.

`deleteFeedItem(communityId, feedItemId)`

Delete a feed item.

API Version

28.0–31.0

⚠️ Important: In version 32.0 and later, use `deleteFeedElement(communityId, feedElementId)`. 

758
Requires Chatter

Yes

Signature

```java
public static Void deleteFeedItem(String communityId, String feedItemId)
```

Parameters

- `communityId`
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- `feedItemId`
  - Type: String
  - ID for a feed item.

Return Value

Type: Void

`getCommentsForFeedItem(communityId, feedItemId)`

Get comments for a feed item.

API Version

28.0–31.0

⚠️ Important: In version 32.0 and later, use `getCommentsForFeedElement(communityId, feedElementId)`.

Available to Guest Users

31.0 only

Requires Chatter

Yes

Signature

```java
public static ConnectApi.CommentPage getCommentsForFeedItem(String communityId, String feedItemId)
```

Parameters

- `communityId`
  - Type: String
  - ID for an Experience Cloud site, internal, or null.
feedItemId
Type: String
ID for a feed item.

Return Value
Type: ConnectApi.CommentPage

getCommentsForFeedItem(communityId, feedItemId, pageParam, pageSize)
Get a page of comments for a feed item.

API Version
28.0–31.0

⚠️ Important: In version 32.0 and later, use getCommentsForFeedElement(communityId, feedElementId, pageParam, pageSize).

Available to Guest Users
31.0 only

Requires Chatter
Yes

Signature
public static ConnectApi.CommentPage getCommentsForFeedItem(String communityId, String feedItemId, String pageParam, Integer pageSize)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedItemId
Type: String
ID for a feed item.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.
Return Value
Type: `ConnectApi.CommentPage`

`getFeedItem(communityId, feedItemId)`
Get a feed item.

API Version
28.0–31.0

⚠️ **Important:** In version 32.0 and later, use `getFeedElement(communityId, feedElementId)`.

Available to Guest Users
31.0 only

Requires Chatter
Yes

Signature
```
public static ConnectApi.FeedItem getFeedItem(String communityId, String feedItemId)
```

Parameters
- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.
- `feedItemId`
  Type: `String`
  ID for a feed item.

Return Value
Type: `ConnectApi.FeedItem`

⚠️ **Note:** Triggers on FeedItem objects run before their attachment and capabilities information is saved, which means that ConnectApi.FeedItem.attachment information and ConnectApi.FeedElement.capabilities information may not be available in the trigger.

`getFeedItemBatch(communityId, feedItemIds)`
Get a list of feed items.

API Version
31.0–31.0
Important: In version 32.0 and later, use `getFeedElementBatch(communityId, feedElementIds)`.

Requires Chatter
Yes

Signature
```
public static ConnectApi.BatchResult[] getFeedItemBatch(String communityId, List<String> feedItemIds)
```

Parameters
- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **feedItemIds**
  - Type: `List<String>`
  - A list of up to 500 feed item IDs.

Return Value
- Type: `ConnectApi.BatchResult[]`

The `ConnectApi.BatchResult.getResult()` method returns a `ConnectApi.FeedItem` object and errors for feed items that didn’t load.

Example
```
// Create a list of feed items.
ConnectApi.FeedItemPage feedItemPage = ConnectApi.ChatterFeeds.getFeedItemsFromFeed(null, ConnectApi.FeedType.Company);
System.debug(feedItemPage);

// Create a list of feed item IDs.
List<String> feedItemIds = new List<String>();
for (ConnectApi.FeedItem aFeedItem : feedItemPage.items){
  feedItemIds.add(aFeedItem.id);
}

// Get info about the feed items in the list.
ConnectApi.BatchResult[] batchResults = ConnectApi.ChatterFeeds.getFeedItemBatch(null, feedItemIds);

for (ConnectApi.BatchResult batchResult : batchResults) {
  if (batchResult.isSuccess()) {
    // Operation was successful.
    // Print the header for each feed item.
    ConnectApi.FeedItem aFeedItem;
    if(batchResult.getResult() instanceof ConnectApi.FeedItem) {
      aFeedItem = (ConnectApi.FeedItem) batchResult.getResult();
      // Further processing of the feed item.
    }
  }
```

```
getFeedItemsFromFeed(communityId, feedType)
Get feed items from the Company, Home, and Moderation feeds.

API Version
28.0–31.0

⚠️ Important: In version 32.0 and later, use getFeedElementsFromFeed(communityId, feedType).

Available to Guest Users
31.0 only

Requires Chatter
Yes

Signature
public static ConnectApi.FeedItemPage getFeedItemsFromFeed(String communityId, ConnectApi.FeedType feedType)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Type of feed. Valid values are Company, DirectMessageModeration, DirectMessages, Home, Moderation, and PendingReview.

Return Value
Type: ConnectApi.FeedItemPage
Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
  setTestGetFeedItemsFromFeed(communityId, feedType, result)

getFeedItemsFromFeed(communityId, feedType, pageParam, pageSize, sortParam)
Get a page of sorted feed items from the Company, Home, and Moderation feeds.

API Version
28.0–31.0
⚠️ Important: In version 32.0 and later, use getFeedElementsFromFeed(communityId, feedType, pageParam, pageSize, sortParam).

Available to Guest Users
31.0 only

Requires Chatter
Yes

Signature
public static ConnectApi.FeedItemPage getFeedItemsFromFeed(String communityId, ConnectApi.FeedType feedType, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam)

Parameters
communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

feedType
  Type: ConnectApi.FeedType
  Type of feed. Valid values are Company, DirectMessageModeration, DirectMessages, Home, Moderation, and PendingReview.

pageParam
  Type: String
  Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
  Type: Integer
Number of feed items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**
- **Type:** `ConnectApi.FeedSortOrder`
- **Values are:**
  - `CreatedDateAsc`—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
  - `CreatedDateDesc`—Sorts by most recent creation date.
  - `LastModifiedDateDesc`—Sorts by most recent activity.
  - `MostViewed`—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
  - `Relevance`—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.

Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in `null`, the default value `CreatedDateDesc` is used.

**Return Value**
- **Type:** `ConnectApi.FeedItemPage`

**Usage**
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

**SEE ALSO:**
- `setTestGetFeedItemsFromFeed(communityId, feedType, pageParam, pageSize, sortParam, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

**getFeedItemsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam)**
Get a page of sorted feed items from the `Company`, `Home`, and `Moderation` feeds. Each feed item contains no more than the specified number of comments.

**API Version**
- 29.0–31.0

**Important:** In version 32.0 and later, use `getFeedElementsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam)`.

**Available to Guest Users**
- 31.0 only

**Requires Chatter**
- Yes
public static ConnectApi.FeedItemPage getFeedItemsFromFeed(String communityId, ConnectApi.FeedType feedType, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Type of feed. Valid values are Company, DirectMessageModeration, DirectMessages, Home, Moderation, and PendingReview.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed item. The default value is 3.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null, the default value CreatedDateDesc is used.

Return Value
Type: ConnectApi.FeedItemPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetFeedItemsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, result)


getFeedItemsFromFeed(communityId, feedType, subjectId)
Get feed items from a feed for a user or record.

API Version
28.0–31.0

Important: In version 32.0 and later, use getFeedElementsFromFeed(communityId, feedType, subjectId).

Available to Guest Users
31.0 only

Requires Chatter
Yes

Signature
public static ConnectApi.FeedItemPage getFeedItemsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Type of feed. Valid values include every ConnectApi.FeedType except Company, DirectMessageModeration, DirectMessages, Filter, Home, Landing, Moderation, and PendingReview.
subjectId
Type: String

If feedType is Record, subjectId can be any record ID, including a group ID. If feedType is Streams, subjectId must be a stream ID. If feedType is Topics, subjectId must be a topic ID. If feedType is UserProfile, subjectId can be any user ID. If the feedType is any other value, subjectId must be the ID of the context user or the alias me.

Return Value
Type: ConnectApi.FeedItemPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetFeedItemsFromFeed(communityId, feedType, subjectId, result)

getFeedItemsFromFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam)
Get a page of sorted feed items from a feed for a user or record.

API Version
28.0–31.0

Important: In version 32.0 and later, use getFeedElementsFromFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam).

Available to Guest Users
31.0 only

Requires Chatter
Yes

Signature
public static ConnectApi.FeedItemPage getFeedItemsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

**feedType**
Type: `ConnectApi.FeedType`
Type of feed. Valid values include every `ConnectApi.FeedType` except Company, DirectMessageModeration, DirectMessages, Filter, Home, Landing, Moderation, and PendingReview.

**subjectId**
Type: `String`
If `feedType` is Record, `subjectId` can be any record ID, including a group ID. If `feedType` is Streams, `subjectId` must be a stream ID. If `feedType` is Topics, `subjectId` must be a topic ID. If `feedType` is UserProfile, `subjectId` can be any user ID. If the `feedType` is any other value, `subjectId` must be the ID of the context user or the alias `me`.

**pageParam**
Type: `String`
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**
Type: `Integer`
Number of feed items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**
Type: `ConnectApi.FeedSortOrder`
Values are:
- **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for Home feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- **Relevance**—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in `null`, the default value `CreatedDateDesc` is used.

**Return Value**
Type: `ConnectApi.FeedItemPage`

**Usage**
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

**SEE ALSO:**
- `setTestGetFeedItemsFromFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam, result)`
getFeedItemsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam)

Get a page of sorted feed items from a feed for a user or record. Each feed item includes no more than the specified number of comments.

API Version
29.0–31.0

⚠️ Important: In version 32.0 and later, use getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam).

Available to Guest Users
31.0 only

Requires Chatter
Yes

Signature
public static ConnectApi.FeedItemPage getFeedItemsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Type of feed. Valid values include every ConnectApi.FeedType except Company, DirectMessageModeration, DirectMessages, Filter, Home, Landing, Moderation, and PendingReview.

subjectId
Type: String
If feedType is Record, subjectId can be any record ID, including a group ID. If feedType is Streams, subjectId must be a stream ID. If feedType is Topics, subjectId must be a topic ID. If feedType is UserProfile, subjectId can be any user ID. If the feedType is any other value, subjectId must be the ID of the context user or the alias me.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed item. The default value is 3.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
• AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.

• FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
• CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
• CreatedDateDesc—Sorts by most recent creation date.
• LastModifiedDateDesc—Sorts by most recent activity.
• MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
• Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null, the default value CreatedDateDesc is used.

Return Value
Type: ConnectApi.FeedItemPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetFeedItemsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, result)

gGetFeedItemsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, showInternalOnly)

Get a page of sorted feed items from a record feed for a user or record. Each feed item includes no more than the specified number of comments. Specify whether to return feed items posted by internal (non-Experience Cloud site) users only.
API Version

30.0–31.0

⚠️ Important: In version 32.0 and later, use `getFeedElementsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, showInternalOnly)`.

Available to Guest Users

31.0 only

Requires Chatter

Yes

Signature

```java
public static ConnectApi.FeedItemPage getFeedItemsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, Boolean showInternalOnly)
```

Parameters

`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

`feedType`
Type: `ConnectApi.FeedType`
Value must be `ConnectApi.FeedType.Record`.

`subjectId`
Type: `String`
Any record ID, including a group ID.

`recentCommentCount`
Type: `Integer`
Maximum number of comments to return with each feed item. The default value is 3.

`density`
Type: `ConnectApi.FeedDensity`
Specify the amount of content in a feed.
- **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

`pageParam`
Type: `String`
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

`pageSize`
Type: `Integer`
Number of feed items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

`sortParam`
Type: `ConnectApi.FeedSortOrder`
Values are:
- `CreatedDateAsc`—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- `CreatedDateDesc`—Sorts by most recent creation date.
- `LastModifiedDateDesc`—Sorts by most recent activity.
- `MostViewed`—Sorts by most viewed content. This sort order is available only for Home feeds when the `ConnectApi.FeedFilter` is UnansweredQuestions.
- `Relevance`—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in `null`, the default value `CreatedDateDesc` is used.

`showInternalOnly`
Type: `Boolean`
Specifies whether to show only feed items from internal (non-Experience Cloud site) users (`true`), or not (`false`). The default value is `false`.

**Return Value**
Type: `ConnectApi.FeedItemPage`

**Usage**
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

**SEE ALSO:**
- `setTestGetFeedItemsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, showInternalOnly, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

**getFeedItemsFromFilterFeed(communityId, subjectId, keyPrefix)**
Get feed items from a feed filtered by a key prefix for a user.

**API Version**
28.0–31.0

**Important:** In version 32.0 and later, use `getFeedElementsFromFilterFeed(communityId, subjectId, keyPrefix)`.
Requires Chatter
Yes

Signature

```java
public static ConnectApi.FeedItemPage getFeedItemsFromFilterFeed(String communityId, String subjectId, String keyPrefix)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- `subjectId`
  - Type: `String`
  - ID of the context user or the alias `me`.

- `keyPrefix`
  - Type: `String`
  - A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

Return Value

Type: `ConnectApi.FeedItemPage`

Usage

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- `setTestGetFeedItemsFromFilterFeed(communityId, subjectId, keyPrefix, result)`

```java
getFeedItemsFromFilterFeed(communityId, subjectId, keyPrefix, pageParam, pageSize, sortParam)
```

Get a page of sorted feed items from a feed filtered by a key prefix for a user.

API Version

28.0–31.0

**Important:** In version 32.0 and later, use `getFeedElementsFromFilterFeed(communityId, subjectId, keyPrefix, pageParam, pageSize, sortParam)`.
Requires Chatter

Yes

Signature

```
public static ConnectApi.FeedItemPage getFeedItemsFromFilterFeed(String communityId, String subjectId, String keyPrefix, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam)
```

Parameters

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- **subjectId**
  - Type: String
  - ID of the context user or the alias me.

- **keyPrefix**
  - Type: String
  - A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

- **pageParam**
  - Type: String
  - Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

- **pageSize**
  - Type: Integer
  - Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

- **sortParam**
  - Type: ConnectApi.FeedSortOrder
  - Values are:
    - CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
    - CreatedDateDesc—Sorts by most recent creation date.
    - LastModifiedDateDesc—Sorts by most recent activity.
    - MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
    - Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
  
  Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null, the default value CreatedDateDesc is used.

Return Value

Type: ConnectApi.FeedItemPage
Usage

To test code that uses this method, use the matching set test method (prefix the method name with \texttt{setTest}). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

\begin{itemize}
  \item \texttt{setTestGetFeedItemsFromFilterFeed(communityId, subjectId, keyPrefix, pageParam, pageSize, sortParam, result)}
  \item \textit{Apex Developer Guide: Testing ConnectApi Code}
\end{itemize}

\texttt{getFeedItemsFromFilterFeed(communityId, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, sortParam)}

Get a page of sorted feed items from a feed filtered by a key prefix for a user. Each feed item contains no more than the specified number of comments.

API Version

29.0–31.0

\begin{itemize}
  \item \textbf{Important:} In version 32.0 and later, use \texttt{getFeedElementsFromFilterFeed(communityId, subjectId, keyPrefix, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, sortParam)}.
\end{itemize}

Requires Chatter

Yes

Signature

\begin{verbatim}
public static ConnectApi.FeedItemPage getFeedItemsFromFilterFeed(String communityId, String subjectId, String keyPrefix, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam)
\end{verbatim}

Parameters

\begin{itemize}
  \item \textit{communityId} \\
      Type: \texttt{String} \\
      ID for an Experience Cloud site, internal, or \texttt{null}.
  \item \textit{subjectId} \\
      Type: \texttt{String} \\
      ID of the context user or the alias \texttt{me}.
  \item \textit{keyPrefix} \\
      Type: \texttt{String} \\
      A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.
  \item \textit{recentCommentCount} \\
      Type: \texttt{Integer} \\
      Maximum number of comments to return with each feed item. The default value is 3.
\end{itemize}
density
  Type: ConnectApi.FeedDensity
  Specify the amount of content in a feed.
  • AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
  • FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
  Type: String
  Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
  Type: Integer
  Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
  Type: ConnectApi.FeedSortOrder
  Values are:
  • CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
  • CreatedDateDesc—Sorts by most recent creation date.
  • LastModifiedDateDesc—Sorts by most recent activity.
  • MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
  • Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
  Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null, the default value CreatedDateDesc is used.

Return Value
  Type: ConnectApi.FeedItemPage

Usage
  To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
  setTestGetFeedItemsFromFilterFeed(communityId, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, sortParam, result)
getFeedItemsFromFilterFeedUpdatedSince(communityId, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, updatedSince)

Get a page of feed items from a feed filtered by a key prefix for a user. Include only feed items that have been updated since the time specified in the updatedSince parameter.

API Version
30.0–31.0

Important: In version 32.0 and later, use getFeedElementsFromFilterFeedUpdatedSince(communityId, subjectId, keyPrefix, recentCommentCount, elementsPerBundle, density, pageParam, pageSize, updatedSince).

Requires Chatter
Yes

Signature
public static ConnectApi.FeedItemPage getFeedItemsFromFilterFeedUpdatedSince(String communityId, String subjectId, String keyPrefix, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, String updatedSince)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subjectId
Type: String
ID of the context user or the alias me.

keyPrefix
Type: String
A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed item. The default value is 3.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.

- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.
**getPageParam**

*Type: String*

Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

**pageSize**

*Type: Integer*

Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

**updatedSince**

*Type: String*

Opaque token containing information about the last modified date of the feed. Do not construct this token. To retrieve this token, call getFeedItemsFromFilterFeed and take the value from the updatesToken property of the ConnectApi.FeedItemPage response body.

**Return Value**

*Type: ConnectApi.FeedItemPage*

**Usage**

This method returns only feed items that have been updated since the time specified in the updatedSince argument. A feed item is considered to be updated if it was created since the last feed request, or if sort=LastModifiedDateDesc and a comment was added to the feed item since the last feed request. Adding likes and topics doesn’t update a feed item.

To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

**SEE ALSO:**

setTestGetFeedItemsFromFilterFeedUpdatedSince(communityId, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, sortParam, updatedSince, result)


**getFeedItemsUpdatedSince(communityId, feedType, recentCommentCount, density, pageParam, pageSize, updatedSince)**

Get a page of feed items from the Company, Home, and Moderation feeds. Include only feed items that have been updated since the time specified in the updatedSince parameter. Each feed item contains no more than the specified number of comments.

**API Version**

30.0–31.0

**Important:** In version 32.0 and later, use getFeedElementsUpdatedSince(communityId, feedType, recentCommentCount, density, pageParam, pageSize, updatedSince).

**Available to Guest Users**

31.0 only
Requires Chatter

Yes

Signature

```java
public static ConnectApi.FeedItemPage getFeedItemsUpdatedSince(String communityId,
    ConnectApi.FeedType feedType, Integer recentCommentCount, ConnectApi.FeedDensity density,
    String pageParam, Integer pageSize, String updatedSince)
```

Parameters

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **feedType**
  - Type: `ConnectApi.FeedType`
  - Type of feed. Valid values are `Company`, `DirectMessageModeration`, `DirectMessages`, `Home`, `Moderation`, and `PendingReview`.

- **recentCommentCount**
  - Type: `Integer`
  - Maximum number of comments to return with each feed item. The default value is 3.

- **density**
  - Type: `ConnectApi.FeedDensity`
  - Specify the amount of content in a feed.
    - **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
    - **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

- **pageParam**
  - Type: `String`
  - Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

- **pageSize**
  - Type: `Integer`
  - Number of feed items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

- **updatedSince**
  - Type: `String`
  - An opaque token containing information about the last modified date of the feed. Do not construct this token. Retrieve this token from the `updateToken` property of the `ConnectApi.FeedItemPage` response body.

Return Value

- Type: `ConnectApi.FeedItemPage`
Usage

This method returns only feed items that have been updated since the time specified in the `updatedSince` argument. A feed item is considered to be updated if it was created since the last feed request, or if `sort=LastModifiedDateDesc` and a comment was added to the feed item since the last feed request. Adding likes and topics doesn’t update a feed item.

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

Example

This example gets the feed items in the company feed and grabs the `updatesToken` property from the returned object. It then passes the value of `updatesToken` to the `getFeedItemsUpdatedSince` method to get the feed items updated since the first call:

```java
// Get the feed items in the company feed and return the updatesToken
String communityId = null;

// Get the feed and extract the update token
ConnectApi.FeedItemPage page = ConnectApi.ChatterFeeds.getFeedItemsFromFeed(communityId, ConnectApi.FeedType.Company);

// page.updatesToken is opaque and has a value like '2:1384549034000'

// Get the feed items that changed since the provided updatesToken
ConnectApi.FeedItemPage feedItems= ConnectApi.ChatterFeeds.getFeedItemsUpdatedSince(communityId, ConnectApi.FeedType.Company, 1, ConnectApi.FeedDensity.AllUpdates, null, 1, page.updatesToken);
```

SEE ALSO:

`setTestGetFeedItemsUpdatedSince(communityId, feedType, recentCommentCount, density, pageParam, pageSize, updatedSince, ConnectApi.FeedItemPage, results)`

*Apex Developer Guide: Testing ConnectApi Code*

`getFeedItemsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince)`

Get a page of feed items from the Files, Groups, News, People, and Record feeds. Include only feed items that have been updated since the time specified in the `updatedSince` parameter. Each feed item contains no more than the specified number of comments.

**API Version**

30.0–31.0

⚠️ **Important:** In version 32.0 and later, use `getFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince)`. 

**Available to Guest Users**

31.0 only
Requires Chatter
Yes

Signature

```java
public static ConnectApi.FeedItemPage getFeedItemsUpdatedSince(String communityId,
ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount,
ConnectApi.FeedDensity density, String pageParam, Integer pageSize, String updatedSince)
```

Parameters

`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

`feedType`
Type: `ConnectApi.FeedType`
One of these values:
- Files
- Groups
- News
- People
- Record

`subjectId`
Type: `String`
If `feedType` is `ConnectApi.Record`, `subjectId` can be any record ID, including a group ID. Otherwise, it must be the context user or the alias `me`.

`recentCommentCount`
Type: `Integer`
Maximum number of comments to return with each feed item. The default value is 3.

`density`
Type: `ConnectApi.FeedDensity`
Specify the amount of content in a feed.
- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

`pageParam`
Type: `String`
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

`pageSize`
Type: `Integer`
Number of feed items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.
updatedSince
Type: String
An opaque token containing information about the last modified date of the feed. Do not construct this token. Retrieve this token from the updatesToken property of the ConnectApi.FeedItemPage response body.

Return Value
Type: ConnectApi.FeedItemPage

Usage
This method returns only feed items that have been updated since the time specified in the updatedSince argument. A feed item is considered to be updated if it was created since the last feed request, or if sort=LastModifiedDateDesc and a comment was added to the feed item since the last feed request. Adding likes and topics doesn’t update a feed item.
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

Example
This example gets the feed items in the news feed and grabs the updatesToken property from the returned object. It then passes the value of updatesToken to the getFeedItemsUpdatedSince method to get the feed items updated since the first call:

```java
// Get the feed items in the news feed and return the updatesToken
String communityId = null;
String subjectId = 'me';

// Get the feed and extract the update token
ConnectApi.FeedItemPage page = ConnectApi.ChatterFeeds.getFeedItemsFromFeed(communityId, ConnectApi.FeedType.News, subjectId);

// page.updatesToken is opaque and has a value like '2:1384549034000'

// Get the feed items that changed since the provided updatesToken
ConnectApi.FeedItemPage feedItems= ConnectApi.ChatterFeeds.getFeedItemsUpdatedSince(communityId, ConnectApi.FeedType.News, subjectId, null, 1, ConnectApi.FeedDensity.AllUpdates, page.updatesToken);
```

SEE ALSO:
setTestGetFeedItemsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince, result)


getFeedItemsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince, showInternalOnly)

Get a page of feed items from a record feed. Include only feed items that have been updated since the time specified in the updatedSince parameter. Specify whether to return feed items posted by internal (non-Experience Cloud site) users only.
API Version
30.0–31.0

⚠ Important: In version 32.0 and later, use getFeedElementsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince, showInternalOnly).

Available to Guest Users
31.0 only

Requires Chatter
Yes

Signature

```java
public static ConnectApi.FeedItemPage getFeedItemsUpdatedSince(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, String updatedSince, Boolean showInternalOnly)
```

Parameters

- **communityId**
  Type: String
  ID for an Experience Cloud site, internal, or null.

- **feedType**
  Type: ConnectApi.FeedType
  Value must be ConnectApi.FeedType.Record.

- **subjectId**
  Type: String
  Any record ID, including a group ID.

- **recentCommentCount**
  Type: Integer
  Maximum number of comments to return with each feed item. The default value is 3.

- **density**
  Type: ConnectApi.FeedDensity
  Specify the amount of content in a feed.
  - **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
  - **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

- **pageParam**
  Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

**pageSize**
Type: Integer
Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

**updatedSince**
Type: String
An opaque token containing information about the last modified date of the feed. Do not construct this token. Retrieve this token from the updatesToken property of the ConnectApi.FeedItemPage response body.

**showInternalOnly**
Type: Boolean
Specifies whether to show only feed items from internal (non-Experience Cloud site) users (true), or not (false). The default value is false.

**Return Value**
Type: `ConnectApi.FeedItemPage`

**Usage**
This method returns only feed items that have been updated since the time specified in the updatedSince argument. A feed item is considered to be updated if it was created since the last feed request, or if sort=LastModifiedDateDesc and a comment was added to the feed item since the last feed request. Adding likes and topics doesn’t update a feed item.

If `showInternalOnly` is true and digital experiences is enabled, feed items from Experience Cloud sites are included. Otherwise, only feed items from the internal site are included.

To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

**Example**
This example gets the feed items in the news feed and grabs the updatesToken property from the returned object. It then passes the value of updatesToken to the getFeedItemsUpdatedSince method to get the feed items updated since the first call:

```java
// Get the feed items in the news feed and return the updatesToken
String communityId = null;
String subjectId = 'me';

// Get the feed and extract the update token
ConnectApi.FeedItemPage page = ConnectApi.ChatterFeeds.getFeedItemsFromFeed(communityId, ConnectApi.FeedType.News, subjectId);

// page.updatesToken is opaque and has a value like '2:1384549034000'

// Get the feed items that changed since the provided updatesToken
ConnectApi.FeedItemPage feedItems= ConnectApi.ChatterFeeds.getFeedItemsUpdatedSince
```
SEE ALSO:

setTestGetFeedItemsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince, showInternalOnly, result)


getFeedPoll(communityId, feedItemID)

Get the poll associated with a feed item.

API Version

28.0–31.0

**Important:** In version 32.0 and later, use getFeedElementPoll(communityId, feedElementId).

Requires Chatter

Yes

**Signature**

```java
public static ConnectApi.FeedPoll getFeedPoll(String communityId, String feedItemId)
```

**Parameters**

- **communityId**
  
  Type: String
  
  ID for an Experience Cloud site, internal, or null.

- **feedItemId**
  
  Type: String
  
  ID for a feed item.

**Return Value**

Type: ConnectApi.FeedPoll

**Note:** Triggers on FeedItem objects run before their attachment and capabilities information is saved, which means that ConnectApi.FeedItem.attachment information and ConnectApi.FeedElement.capabilities information may not be available in the trigger.

getLikesForFeedItem(communityId, feedItemID)

Get likes for a feed item.
API Version
28.0–31.0

Important: In version 32.0 and later, use `getLikesForFeedElement(communityId, feedElementId)`.

Available to Guest Users
31.0 only

Requires Chatter
Yes

Signature
```java
public static ConnectApi.ChatterLikePage getLikesForFeedItem(String communityId, String feedItemId)
```

Parameters
`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

`feedItemId`
Type: `String`
ID for a feed item.

Return Value
Type: `ConnectApi.ChatterLikePage`

`getLikesForFeedItem(communityId, feedItemId, pageParam, pageSize)`
Get a page of likes for a feed item.

API Version
28.0–31.0

Important: In version 32.0 and later, use `getLikesForFeedElement(communityId, feedElementId, pageParam, pageSize)`.

Available to Guest Users
31.0 only

Requires Chatter
Yes
Signature

```java
public static ConnectApi.ChatterLikePage getLikesForFeedItem(String communityId, String feedItemId, Integer pageParam, Integer pageSize)
```

Parameters

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- **feedItemId**
  - Type: String
  - ID for a feed item.

- **pageParam**
  - Type: Integer
  - Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

- **pageSize**
  - Type: Integer
  - Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

Return Value

- Type: `ConnectApi.ChatterLikePage`

**likeFeedItem(communityId, feedItemId)**

Like a feed item for the context user.

API Version

28.0–31.0

ℹ️ Important: In version 32.0 and later, use `likeFeedElement(communityId, feedElementId)`.

Requires Chatter

Yes

Signature

```java
public static ConnectApi.ChatterLike likeFeedItem(String communityId, String feedItemId)
```

Parameters

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.
feedItemId
Type: String
ID for a feed item.

Return Value
Type: ConnectApi.ChatterLike
If the context user already liked the feed item, this method is a non-operation and returns the existing like.

postComment(communityId, feedItemId, text)
Post a plain-text comment to a feed item.

API Version
28.0–31.0

Important: In version 32.0 and later, use postCommentToFeedElement(communityId, feedElementId, text).

Requires Chatter
Yes

Signature
public static ConnectApi.Comment postComment(String communityId, String feedItemId, String text)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedItemId
Type: String
ID for a feed item.

text
Type: String
The text of the comment. Mentions are downgraded to plain text. To include a mention that links to a user, call postComment(communityId, feedItemId, comment, feedItemFileUpload) and pass the mention in a ConnectApi.CommentInput object.

Return Value
Type: ConnectApi.Comment
Usage

If hashtags or links are detected in text, they're included in the comment as hashtag and link segments. Mentions aren't detected in text and aren't separated out of the text.

Feed items and comments can contain up to 10,000 characters.

`postComment(communityId, feedItemId, comment, feedItemFileUpload)`

Post a rich-text comment to a feed item. Use this method to include mentions and to attach a file to a comment.

API Version

28.0–31.0

⚠️ Important: In version 32.0 and later, use `postCommentToFeedElement(communityId, feedElementId, comment, feedElementFileUpload)`.

Requires Chatter

Yes

Signature

```java
public static ConnectApi.Comment postComment(String communityId, String feedItemId, ConnectApi.CommentInput comment, ConnectApi.BinaryInput feedItemFileUpload)
```

Parameters

- `communityId`
  
  Type: `String`
  
  ID for an Experience Cloud site, internal, or `null`.

- `feedItemId`
  
  Type: `String`
  
  ID for a feed item.

- `comment`
  
  Type: `ConnectApi.CommentInput`
  
  In the `CommentInput` object, specify rich text, including @mentions. Optionally, in the `CommentInput.attachment` property, specify an existing file or a new file

- `feedItemFileUpload`
  
  Type: `ConnectApi.BinaryInput`
  
  If you specify a `NewFileAttachmentInput` object in the `CommentInput.attachment` property, specify the new binary file to attach in this argument. Otherwise, do not specify a value.

Return Value

Type: `ConnectApi.Comment`
Usage

Feed items and comments can contain up to 10,000 characters.

Sample: Posting a Comment with a New File Attachment

To post a comment and upload and attach a new file to the comment, create a `ConnectApi.CommentInput` object and a `ConnectApi.BinaryInput` object to pass to the `ConnectApi.ChatterFeeds.postComment` method.

```java
String communityId = null;
String feedItemId = '0D5D0000000Kcd1';

ConnectApi.CommentInput input = new ConnectApi.CommentInput();
ConnectApi.TextSegmentInput textSegment;

String commentId = null;
String feedItemId = '0D5D0000000Kcd1';

ConnectApi.CommentInput input = new ConnectApi.CommentInput();
ConnectApi.TextSegmentInput textSegment;

ConnectApi.NewFileAttachmentInput attachmentInput = new ConnectApi.NewFileAttachmentInput();
attachmentInput.description = 'The description of the file';
attachmentInput.title = 'contentFile.txt';
input.attachment = attachmentInput;

String fileContents = 'This is the content of the file.';
Blob fileBlob = Blob.valueOf(fileContents);

ConnectApi.Comment commentRep = ConnectApi.ChatterFeeds.postComment(communityId, feedItemId, input, binaryInput);
```

`postFeedElement(communityId, feedElement, feedElementFileUpload)`

Post a rich-text feed element. Include mentions and hashtag topics, attach a file to a feed element, and associate action link groups with a feed element. You can also use this method to share a feed element and add a comment.

API Version

31.0–35.0

⚠️ Important: In version 36.0 and later, this method is no longer available because you can’t create a feed post and upload a binary file in the same call. Upload files to Salesforce first, and then use `postFeedElement(communityId, feedElement)` to create the feed post and attach the files.

Requires Chatter

Yes
Signature

```java
public static ConnectApi.FeedElement postFeedElement(String communityId,
ConnectApi.FeedElementInput feedElement, ConnectApi.BinaryInput feedElementFileUpload)
```

Parameters

communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

feedElement
  Type: ConnectApi.FeedElementInput
  Specify rich text, including mentions. Optionally, specify a link, a poll, an existing file, or a new file.

feedElementFileUpload
  Type: ConnectApi.BinaryInput
  Specify the new binary file to attach to the post only if you also specify a NewFileAttachmentInput object in the feedElement parameter. Otherwise, pass null.

Return Value

Type: ConnectApi.FeedElement

Example for Posting a Feed Element with a New (Binary) File

```java
ConnectApi.FeedItemInput input = new ConnectApi.FeedItemInput();
input.subjectId = 'me';

ConnectApi.ContentCapabilityInput contentInput = new ConnectApi.ContentCapabilityInput();
contentInput.title = 'Title';

ConnectApi.FeedElementCapabilitiesInput capabilities = new
ConnectApi.FeedElementCapabilitiesInput();
capabilities.content = contentInput;

capabilities.content = contentInput;

capabilities = capabilities;

String text = 'These are the contents of the new file.';
Blob myBlob = Blob.valueOf(text);
ConnectApi.BinaryInput binInput = new ConnectApi.BinaryInput(myBlob, 'text/plain', 'fileName');

ConnectApi.ChatterFeeds.postFeedElement(Network.getNetworkId(), input, binInput);
```

postFeedItem(communityId, feedType, subjectId, text)
Post a plain-text feed item.

API Version

28.0–31.0
Important: In version 32.0 and later, use `postFeedElement(communityId, subjectId, feedElementType, text)`.

Requires Chatter
Yes

Signature
```
public static ConnectApi.FeedItem postFeedItem(String communityId, ConnectApi.FeedType feedType, String subjectId, String text)
```

Parameters
- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.
- **feedType**
  - Type: `ConnectApi.FeedType`
  - One of the following:
    - News
    - Record
    - UserProfile
  - Use Record to post to a group.
- **subjectId**
  - Type: String
  - The value depends on the `feedType`:
    - News—ID of the context user or the keyword me.
    - Record—ID of any record with a feed, including groups.
    - UserProfile—ID of any user.
- **text**
  - Type: String
  - Text of the feed item. Mentions are downgraded to plain text. To include a mention that links to the user, call the `postFeedItem(communityId, feedType, subjectId, feedItemInput, feedItemFileUpload)` method and pass the mention in a `ConnectApi.FeedItemInput` object.

Return Value
- Type: `ConnectApi.FeedItem`

Note: Triggers on FeedItem objects run before their attachment and capabilities information is saved, which means that `ConnectApi.FeedItem.attachment` information and `ConnectApi.FeedElement.capabilities` information may not be available in the trigger.
Usage

Feed items and comments can contain up to 10,000 characters.

Posts to ConnectApi.FeedType.UserProfile in API versions 23.0 and 24.0 created user status updates, not feed items. For posts to the User Profile Feed in those API versions, the character limit is 1,000 characters.

postFeedItem(communityId, feedType, subjectId, feedItemInput, feedItemFileUpload)

Post a rich-text feed item to a feed. Use this method to include mentions and hashtag topics and to attach a file to a feed item. You can also use this method to share a feed item and add a comment.

API Version

28.0–31.0

⚠️ Important: In version 32.0 and later, use postFeedElement(communityId, feedElement, feedElementFileUpload).

Requires Chatter

Yes

Signature

public static ConnectApi.FeedItem postFeedItem(String communityId, ConnectApi.FeedType feedType, String subjectId, ConnectApi.FeedItemInput feedItemInput, ConnectApi.BinaryInput feedItemFileUpload)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
One of the following:
- News
- Record
- UserProfile

To post a feed item to a group, use Record and use a group ID as the subjectId.

subjectId
Type: String

If feedType is Record, subjectId can be any record ID, including a group ID. If feedType is Streams, subjectId must be a stream ID. If feedType is Topics, subjectId must be a topic ID. If feedType is UserProfile, subjectId can be any user ID. If the feedType is any other value, subjectId must be the ID of the context user or the alias me.

feedItemInput
Type: ConnectApi.FeedItemInput
In the FeedItemInput object, specify rich text. Optionally, in the FeedItemInput.attachment property, specify a link, a poll, an existing file, or a new file.

**feedItemFileUpload**
Type: ConnectApi.BinaryInput

If you specify a NewFileAttachmentInput object in the FeedItemInput.attachment property, specify the new binary file to attach in this argument. Otherwise, do not specify a value.

**Return Value**
Type: ConnectApi.FeedItem

*Note:* Triggers on FeedItem objects run before their attachment and capabilities information is saved, which means that ConnectApi.FeedItem.attachment information and ConnectApi.FeedElement.capabilities information may not be available in the trigger.

**Usage**
Feed items and comments can contain up to 10,000 characters. Posts to ConnectApi.FeedType.UserProfile in API versions 23.0 and 24.0 created user status updates, not feed items. For posts to the User Profile Feed in those API versions, the character limit is 1,000 characters.

**Example for Sharing a Feed Item and Adding a Comment**
To share a feed item and add a comment, create a ConnectApi.FeedItemInput object containing the comment and the feed item to share. Then pass the object to ConnectApi.ChatterFeeds.postFeedItem in the feedItemInput argument. The message segments in the message body input are used as the comment.

```java
ConnectApi.FeedItemInput input = new ConnectApi.FeedItemInput();
input.originalFeedItemId = '0D5D0000000JuAG';

List<ConnectApi.MessageSegmentInput> segmentList = new List<ConnectApi.MessageSegmentInput>();
ConnectApi.TextSegmentInput textSegment = new ConnectApi.TextSegmentInput();
textSegment.text = 'I hope you enjoy this post I found in another group.';
segmentList.add((ConnectApi.MessageSegmentInput)textSegment);
body.messageSegments = segmentList;
input.body = body;

ConnectApi.ChatterFeeds.postFeedItem(null, ConnectApi.FeedType.UserProfile, 'me', input, null);
```

**Example for Posting a Mention to a User Profile Feed**
To post to a user profile feed and include an @mention, call the ConnectApi.ChatterFeeds.postFeedItem method.

```java
String communityId = null;
ConnectApi.FeedType feedType = ConnectApi.FeedType.UserProfile;

ConnectApi.FeedItemInput input = new ConnectApi.FeedItemInput();
ConnectApi.TextSegmentInput textSegment;
```
ConnectApi.MentionSegmentInput mentionSegment = new ConnectApi.MentionSegmentInput();
messageInput.messageSegments = new List<ConnectApi.MessageSegmentInput>();
textSegment = new ConnectApi.TextSegmentInput();
textSegment.text = 'Hey there ';
messageInput.messageSegments.add(textSegment);

mentionSegment.id = '005D0000001LLO1';
messageInput.messageSegments.add(mentionSegment);

textSegment = new ConnectApi.TextSegmentInput();
textSegment.text = '. How are you?';
messageInput.messageSegments.add(textSegment);

input.body = messageInput;

ConnectApi.FeedItem feedItemRep = ConnectApi.ChatterFeeds.postFeedItem(communityId, feedType, 'me', input, null);

searchFeedItems(communityId, q)
Get the feed items that match the search criteria.

API Version
28.0–31.0

⚠️ Important: In version 32.0 and later, use searchFeedElements(communityId, q).

Available to Guest Users
31.0 only

Requires Chatter
Yes

Signature
public static ConnectApi.FeedItemPage searchFeedItems(String communityId, String q)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.
Return Value
Type: `ConnectApi.FeedItemPage`

Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestSearchFeedItems(communityId, q, result)`
  - *Apex Developer Guide: Testing ConnectApi Code*

`searchFeedItems(communityId, q, sortParam)`
Get the sorted feed items that match the search criteria.

API Version
28.0–31.0

⚠️ **Important:** In version 32.0 and later, use `searchFeedElements(communityId, q, sortParam)`.

Available to Guest Users
31.0 only

Requires Chatter
Yes

Signature
```java
public static ConnectApi.FeedItemPage searchFeedItems(String communityId, String q,
ConnectApi.FeedSortOrder sortParam)
```

Parameters
- `communityId`
  Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.
- `q`
  Type: `String`
  - Required and can’t be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See *Wildcards*.
- `sortParam`
  Type: `ConnectApi.FeedSortOrder`
  - Values are:
• `CreatedDateAsc`—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
• `CreatedDateDesc`—Sorts by most recent creation date.
• `LastModifiedDateDesc`—Sorts by most recent activity.
• `MostViewed`—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
• `Relevance`—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.

Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in `null`, the default value `CreatedDateDesc` is used.

**Return Value**

Type: `ConnectApi.FeedItemPage`

**Usage**

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

`setTestSearchFeedItems(communityId, q, sortParam, result)`

*Apex Developer Guide: Testing ConnectApi Code*

`searchFeedItems(communityId, q, pageParam, pageSize)`

Get a page of feed items that match the search criteria.

**API Version**

28.0–31.0

⚠️ **Important:** In version 32.0 and later, use `searchFeedElements(communityId, q, pageParam, pageSize)`.

**Available to Guest Users**

31.0 only

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.FeedItemPage searchFeedItems(String communityId, String q, String pageParam, Integer pageSize)
```
Parameters

- **communityId**
  Type: String
  ID for an Experience Cloud site, internal, or null.

- **q**
  Type: String
  Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

- **pageParam**
  Type: String
  Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

- **pageSize**
  Type: Integer
  Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

Return Value

Type: ConnectApi.FeedItemPage

Usage

To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- setTestSearchFeedItems(communityId, q, pageParam, pageSize, result)

**searchFeedItems(communityId, q, pageParam, pageSize, sortParam)**

Get a page of sorted feed items that match the search criteria.

API Version

28.0–31.0

**Important:** In version 32.0 and later, use searchFeedElements(communityId, q, pageParam, pageSize, sortParam).

Available to Guest Users

31.0 only

Requires Chatter

Yes
public static ConnectApi.FeedItemPage searchFeedItems(String communityId, String q, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds. Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null, the default value CreatedDateDesc is used.

Return Value
Type: ConnectApi.FeedItemPage
Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestSearchFeedItems(communityId, q, pageParam, pageSize, sortParam, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

**searchFeedItems(communityId, q, recentCommentCount, pageParam, pageSize, sortParam)**

Get a page of sorted feed items that match the search criteria.

API Version
29.0–31.0

**Important:** In version 32.0 and later, use `searchFeedElements(communityId, q, recentCommentCount, pageParam, pageSize, sortParam)`.

Available to Guest Users
31.0 only

Requires Chatter
Yes

Signature
```java
public static ConnectApi.FeedItemPage searchFeedItems(String communityId, String q, Integer recentCommentCount, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam)
```

Parameters
- **communityId**
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.
- **q**
  Type: `String`
  Required and can't be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See *Wildcards*.
- **recentCommentCount**
  Type: `Integer`
  Maximum number of comments to return with each feed item. The default value is 3.
pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
• CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
• CreatedDateDesc—Sorts by most recent creation date.
• LastModifiedDateDesc—Sorts by most recent activity.
• MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
• Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null, the default value CreatedDateDesc is used.

Return Value
Type: ConnectApi.FeedItemPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestSearchFeedItems(communityId, q, recentCommentCount, pageParam, pageSize, sortParam, result)

searchFeedItemsInFeed(communityId, feedType, q)
Get the feed items from the Company, Home, and Moderation feeds that match the search criteria.

API Version
28.0–31.0

Important: In version 32.0 and later, use searchFeedElementsInFeed(communityId, feedType, q).
Available to Guest Users
31.0 only

Requires Chatter
Yes

Signature

```java
public static ConnectApi.FeedItemPage searchFeedItemsInFeed(String communityId,
    ConnectApi.FeedType feedType, String q)
```

Parameters

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- **feedType**
  - Type: `ConnectApi.FeedType`
  - Type of feed. Valid values are Company, DirectMessageModeration, DirectMessages, Home, Moderation, and PendingReview.

- **q**
  - Type: String
  - Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

Return Value

Type: `ConnectApi.FeedItemPage`

Usage

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- `setTestSearchFeedItemsInFeed(communityId, feedType, q, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

```java
searchFeedItemsInFeed(communityId, feedType, pageParam, pageSize, sortParam, q)
```

Get a page of sorted feed items from the Company, Home, and Moderation feeds that match the search criteria.

API Version

28.0–31.0
Important: In version 32.0 and later, use searchFeedElementsInFeed(communityId, feedType, pageParam, pageSize, sortParam, q).

Available to Guest Users
31.0 only

Requires Chatter
Yes

Signature

public static ConnectApi.FeedItemPage searchFeedItemsInFeed(String communityId, ConnectApi.FeedType feedType, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q)

Parameters

communityId
 Type: String
 ID for an Experience Cloud site, internal, or null.

feedType
 Type: ConnectApi.FeedType
 Type of feed. Valid values are Company, DirectMessageModeration, DirectMessages, Home, Moderation, and PendingReview.

pageParam
 Type: String
 Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
 Type: Integer
 Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
 Type: ConnectApi.FeedSortOrder
 Values are:
 • CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
 • CreatedDateDesc—Sorts by most recent creation date.
 • LastModifiedDateDesc—Sorts by most recent activity.
 • MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
 • Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
 Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null, the default value CreatedDateDesc is used.
q
  Type: String
  Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

Return Value
Type: ConnectApi.FeedItemPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
  setTestSearchFeedItemsInFeed(communityId, feedType, pageParam, pageSize, sortParam, q, result)

searchFeedItemsInFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, q)
Get a page of sorted feed items from the Company, Home, and Moderation feeds that match the search criteria. Each feed item includes no more than the specified number of comments.

API Version
29.0–31.0

Important: In version 32.0 and later, use searchFeedElementsInFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, q).

Available to Guest Users
31.0 only

Requires Chatter
Yes

Signature
public static ConnectApi.FeedItemPage searchFeedItemsInFeed(String communityId, ConnectApi.FeedType feedType, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q)

Parameters
  communityId
    Type: String
ID for an Experience Cloud site, internal, or null.

`feedType`
Type: `ConnectApi.FeedType`
Type of feed. Valid values are `Company`, `DirectMessageModeration`, `DirectMessages`, `Home`, `Moderation`, and `PendingReview`.

`recentCommentCount`
Type: `Integer`
Maximum number of comments to return with each feed item. The default value is 3.

`density`
Type: `ConnectApi.FeedDensity`
Specify the amount of content in a feed.
- `AllUpdates`—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- `FewerUpdates`—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

`pageParam`
Type: `String`
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

`pageSize`
Type: `Integer`
Number of feed items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

`sortParam`
Type: `ConnectApi.FeedSortOrder`
Values are:
- `CreatedDateAsc`—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- `CreatedDateDesc`—Sorts by most recent creation date.
- `LastModifiedDateDesc`—Sorts by most recent activity.
- `MostViewed`—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- `Relevance`—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.
Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in `null`, the default value `CreatedDateDesc` is used.

`q`
Type: `String`
Required and can’t be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See `Wildcards`.

**Return Value**
Type: `ConnectApi.FeedItemPage`
Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestSearchFeedItemsInFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, q, result)`


`searchFeedItemsInFeed(communityId, feedType, subjectId, q)`
Get the feed items from a feed that match the search criteria.

API Version
28.0–31.0

⚠️ **Important:** In version 32.0 and later, use `searchFeedElementsInFeed(communityId, feedType, subjectId, q)`.

Available to Guest Users
31.0 only

Requires Chatter
Yes

Signature
```
public static ConnectApi.FeedItemPage searchFeedItemsInFeed(String communityId, 
                   ConnectApi.FeedType feedType, String subjectId, String q)
```

Parameters
- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.
- `feedType`
  - Type: `ConnectApi.FeedType`
  - Type of feed. Valid values include every `ConnectApi.FeedType` except Company, DirectMessages, Filter, Landing, and Streams.
- `subjectId`
  - Type: `String`
  - If `feedType` is Record, `subjectId` can be any record ID, including a group ID. If `feedType` is UserProfile, `subjectId` can be any user ID. If `feedType` is any other value, `subjectId` must be the ID of the context user or the alias `me`.
- `q`
  - Type: `String`
Required and can't be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

**Return Value**
Type: `ConnectApi.FeedItemPage`

**Usage**
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

**SEE ALSO:**
- `setTestSearchFeedItemsInFeed(communityId, feedType, subjectId, q, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

### searchFeedItemsInFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam, q)
Get a page of sorted feed items from a feed for a user or record that match the search criteria.

**API Version**
28.0–31.0

⚠️ **Important:** In version 32.0 and later, use `searchFeedElementsInFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam, q)`.

**Available to Guest Users**
31.0 only

**Requires Chatter**
Yes

**Signature**
```java
public static ConnectApi.FeedItemPage searchFeedItemsInFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q)
```

**Parameters**
- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.
- `feedType`
  - Type: `ConnectApi.FeedType`
Type of feed. Valid values include every `ConnectApi.FeedType` except `Company`, `DirectMessages`, `Filter`, `Landing`, and `Streams`.

`subjectId`
Type: String  
If `feedType` is `Record`, `subjectId` can be any record ID, including a group ID. If `feedType` is `Streams`, `subjectId` must be a stream ID. If `feedType` is `Topics`, `subjectId` must be a topic ID. If `feedType` is `UserProfile`, `subjectId` can be any user ID. If the `feedType` is any other value, `subjectId` must be the ID of the context user or the alias `me`.

`pageParam`
Type: String  
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

`pageSize`
Type: Integer  
Number of feed items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

`sortParam`
Type: `ConnectApi.FeedSortOrder`  
Order of feed items in the feed.
- `CreatedByAsc`—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- `CreatedDateDesc`—Sorts by most recent creation date.
- `LastModifiedDateDesc`—Sorts by most recent activity.
- `MostViewed`—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- `Relevance`—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.

Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in `null`, the default value `CreatedDateDesc` is used.

`q`
Type: String  
Search term. Searches keywords in the user or group name. A minimum of one character is required. This parameter doesn’t support wildcards. This parameter is required.

Return Value
Type: `ConnectApi.FeedItemPage`

Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:  
`setTestSearchFeedItemsInFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam, q, result)`  
searchFeedItemsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q)

Get a page of sorted feed items from a feed that match the search criteria. Each feed item includes no more than the specified number of comments.

API Version
29.0–31.0

⚠️ Important: In version 32.0 and later, use searchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q).

Available to Guest Users
31.0 only

Requires Chatter
Yes

Signature
public static ConnectApi.FeedItemPage searchFeedItemsInFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Type of feed. Valid values include every ConnectApi.FeedType except Company, DirectMessages, Filter, Landing, and Streams.

subjectId
Type: String
If feedType is Record, subjectId can be any record ID, including a group ID. If feedType is Streams, subjectId must be a stream ID. If feedType is Topics, subjectId must be a topic ID. If feedType is UserProfile, subjectId can be any user ID. If the feedType is any other value, subjectId must be the ID of the context user or the alias me.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed item. The default value is 3.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
- **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**
Type: **String**
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**
Type: **Integer**
Number of feed items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**
Type: **ConnectApi.FeedSortOrder**
Values are:
- **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for Home feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- **Relevance**—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in `null`, the default value `CreatedDateDesc` is used.

**q**
Type: **String**
Required and can't be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

**Return Value**
Type: **ConnectApi.FeedItemPage**

**Usage**
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

**SEE ALSO:**
- setTestSearchFeedItemsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, result)
- *Apex Developer Guide: Testing ConnectApi Code*
**searchFeedItemsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, showInternalOnly)**

Get a page of sorted feed items from a feed for a user or record that match the search criteria. Each feed item includes no more than the specified number of comments. Specify whether to return feed items posted by internal (non-Experience Cloud site) users only.

**API Version**
30.0–31.0

ℹ️ **Important:** In version 32.0 and later, use `searchFeedElementsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, showInternalOnly)`.

**Available to Guest Users**
31.0 only

**Requires Chatter**
Yes

**Signature**

public static ConnectApi.FeedItemPage searchFeedItemsInFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q, Boolean showInternalOnly)

**Parameters**

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- **feedType**
  - Type: ConnectApi.FeedType
  - Value must be ConnectApi.FeedType.Record.

- **subjectId**
  - Type: String
  - Any record ID, including a group ID.

- **recentCommentCount**
  - Type: Integer
  - Maximum number of comments to return with each feed item. The default value is 3.

- **density**
  - Type: ConnectApi.FeedDensity
  - Specify the amount of content in a feed.
    - AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
• FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
• CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
• CreatedDateDesc—Sorts by most recent creation date.
• LastModifiedDateDesc—Sorts by most recent activity.
• MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
• Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds. Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null, the default value CreatedDateDesc is used.

q
Type: String
Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

showInternalOnly
Type: Boolean
Specifies whether to show only feed items from internal (non-Experience Cloud site) users (true), or not (false). The default value is false.

Return Value
Type: ConnectApi.FeedItemPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestSearchFeedItemsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, showInternalOnly, result)
**searchFeedItemsInFilterFeed**(communityId, subjectId, keyPrefix, q)

Get the feed items that match the search criteria from a feed filtered by a key prefix for a user.

**API Version**

28.0–31.0

⚠️ **Important:** In version 32.0 and later, use `searchFeedElementsInFilterFeed(communityId, subjectId, keyPrefix, q)`.

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.FeedItemPage searchFeedItemsInFilterFeed(String communityId,
    String subjectId, String keyPrefix, String q)
```

**Parameters**

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **subjectId**
  - Type: `String`
  - ID of the context user or the alias `me`.

- **keyPrefix**
  - Type: `String`
  - A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

- **q**
  - Type: `String`
  - Required and can’t be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See [Wildcards](#).

**Return Value**

Type: `ConnectApi.FeedItemPage`
Usage

To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

setTestSearchFeedItemsInFilterFeed(communityId, subjectId, keyPrefix, q, result)


searchFeedItemsInFilterFeed(communityId, subjectId, keyPrefix, pageParam, pageSize, sortParam, q)

Get a page of sorted feed items that match the search criteria from a feed filtered by a key prefix for a user.

API Version

28.0–31.0

Important: In version 32.0 and later, use searchFeedElementsInFilterFeed(communityId, subjectId, keyPrefix, pageParam, pageSize, sortParam, q).

Requires Chatter

Yes

Signature

public static ConnectApi.FeedItemPage searchFeedItemsInFilterFeed(String communityId, String subjectId, String keyPrefix, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q)

Parameters

communityId

Type: String

ID for an Experience Cloud site, internal, or null.

subjectId

Type: String

ID of the context user or the alias me.

keyPrefix

Type: String

A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

pageParam

Type: String

Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.
pageSize
Type: Integer
Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null, the default value CreatedDateDesc is used.

q
Type: String
Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

Return Value
Type: ConnectApi.FeedItemPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- setTestSearchFeedItemsInFilterFeed(communityId, feedType, subjectId, keyPrefix, pageParam, pageSize, sortParam, q, result)

searchFeedItemsInFilterFeed(communityId, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, sortParam, q)

Get a page of sorted feed items that match the search criteria from a feed filtered by a key prefix for a user. Each feed item includes no more than the specified number of comments.

API Version
29.0–31.0

Important: In version 32.0 and later, use searchFeedElementsInFilterFeed(communityId, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, sortParam, q).
Requires Chatter

Yes

Signature

public static ConnectApi.FeedItemPage searchFeedItemsInFilterFeed(String communityId, String subjectId, String keyPrefix, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subjectId
Type: String
ID of the context user or the alias me.

keyPrefix
Type: String
A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed item. The default value is 3.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
• CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.

• CreatedDateDesc—Sorts by most recent creation date.

• LastModifiedDateDesc—Sorts by most recent activity.

• MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.

• Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null, the default value CreatedDateDesc is used.

q
Type: String
Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

Return Value
Type: ConnectApi.FeedItemPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestSearchFeedItemsInFilterFeed(communityId, feedType, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, sortParam, q, result)

shareFeedElement(communityId, subjectId, feedElementType, originalFeedElementId)
Share the originalFeedElementId as the context user.

API Version
31.0–38.0

Important: In version 39.0 and later, use postFeedElement(communityId, feedElement) or updateFeedElement(communityId, feedElementId, feedElement) with the ConnectApi.FeedEntityShareCapabilityInput to share a feed entity with a feed element.

Requires Chatter
Yes
Signature

```java
public static ConnectApi.FeedElement shareFeedElement(String communityId, String subjectId, ConnectApi.FeedElementType feedElementType, String originalFeedElementId)
```

Parameters

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **subjectId**
  - Type: `String`
  - The ID of the user or group with whom to share the feed element.

- **feedElementType**
  - Type: `ConnectApi.FeedElementType`
  - Values are:
    - `Bundle`—A container of feed elements. A bundle also has a body made up of message segments that can always be gracefully degraded to text-only values.
    - `FeedItem`—A feed item has a single parent and is scoped to one Experience Cloud site or across all Experience Cloud sites. A feed item can have capabilities such as bookmarks, canvas, content, comment, link, poll. Feed items have a body made up of message segments that can always be gracefully degraded to text-only values.
    - `Recommendation`—A recommendation is a feed element with a recommendations capability. A recommendation suggests records to follow, groups to join, or applications that are helpful to the context user.

- **originalFeedElementId**
  - Type: `String`
  - The ID of the feed element to share.

Return Value

- Type: `ConnectApi.FeedElement`

Example

```java
ConnectApi.ChatterFeeds.shareFeedElement(null, '0F9RR0000004CPw',
ConnectApi.FeedElementType.FeedItem, '0D5RR0000004Gxc');
```

`shareFeedItem(communityId, feedType, subjectId, originalFeedItemId)`

Share the `originalFeedItemId` to the feed specified by the `feedType`.

API Version

- 28.0–31.0

Important:

- In version 32.0–38.0, use `shareFeedElement(communityId, subjectId, feedElementType, originalFeedElementId)`.
In version 39.0 and later, use `postFeedElement(communityId, feedElement)` or `updateFeedElement(communityId, feedElementId, feedElement)` with the `ConnectApi.FeedEntityShareCapabilityInput`.

**Requires Chatter**
Yes

**Signature**

```java
public static ConnectApi.FeedItem shareFeedItem(String communityId, ConnectApi.FeedType feedType, String subjectId, String originalFeedItemId)
```

**Parameters**

- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.

- `feedType`
  Type: `ConnectApi.FeedType`
  One of the following:
  - News
  - Record
  - UserProfile

  To share a feed item with a group, use `Record` and use a group ID as the `subjectId`.

- `subjectId`
  Type: `String`
  The value depends on the value of `feedType`:
  - News—`subjectId` must be the ID of the context user or the keyword `me`.
  - Record—`subjectId` can be a group ID or the ID of the context user (or `me`).
  - UserProfile—`subjectId` can be any user ID.

- `originalFeedItemId`
  Type: `String`
  The ID of the feed item to share.

**Return Value**

Type: `ConnectApi.FeedItem`
**Example**

To share a feed item with a group, pass in the Experience Cloud site ID (or null), the feed type Record, the group ID, and the ID of the feed item to share.

```java
ConnectApi.ChatterFeeds.shareFeedItem(null, ConnectApi.FeedType.Record, '0F9D00000000izf', '0D5D0000000JuAG');
```

**updateBookmark(communityId, feedItemId, isBookmarkedByCurrentUser)**

Bookmark a feed item or remove a bookmark from a feed item.

**API Version**

28.0–31.0

⚠️ **Important:** In version 32.0 and later, use `updateFeedElementBookmarks(communityId, feedElementId, bookmarks)` or `updateFeedElementBookmarks(communityId, feedElementId, isBookmarkedByCurrentUser)`.

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.FeedItem updateBookmark(String communityId, String feedItemId, Boolean isBookmarkedByCurrentUser)
```

**Parameters**

- `communityId`
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- `feedItemId`
  - Type: String
  - ID for a feed item.

- `isBookmarkedByCurrentUser`
  - Type: Boolean
  - Specifying true adds the feed item to the list of bookmarks for the context user. Specify false to remove a bookmark.

**Return Value**

Type: `ConnectApi.FeedItem`

**voteOnFeedPoll(communityId, feedItemId, myChoiceId)**

Vote or change your vote on a feed poll.
### voteOnFeedPoll

**Signature**

```java
public static ConnectApi.FeedPoll voteOnFeedPoll(String communityId, String feedItemId, String myChoiceId)
```

**Parameters**

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **feedItemId**
  - Type: `String`
  - ID of the feed item that is associated with the poll.

- **myChoiceId**
  - Type: `String`
  - ID of the item in the poll you’re voting for.

**Return Value**

Type: `ConnectApi.FeedPoll`

### setTestGetFeedItemsFromFeed

**Signature**

```java
public static Void setTestGetFeedItemsFromFeed(String communityId, ConnectApi.FeedType feedType, ConnectApi.FeedItemPage result)
```

**Parameters**

- **communityId**
  - Type: `String`

**setTestGetFeedItemsFromFeed(communityId, feedType, result)**

Register a `ConnectApi.FeedItemPage` object to be returned when `getFeedItemsFromFeed` is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

**API Version**

28.0–31.0

**Important:** In version 32.0 and later, use `voteOnFeedElementPoll(communityId, feedElementId, myChoiceId)`.
ID for an Experience Cloud site, internal, or null.

**feedType**

Type: `ConnectApi.FeedType`

Type of feed. Valid values are `Company`, `DirectMessageModeration`, `DirectMessages`, `Home`, `Moderation`, and `PendingReview`.

**result**

Type: `ConnectApi.FeedItemPage`

Object containing test data.

**Return Value**

Type: Void

SEE ALSO:

- `getFeedItemsFromFeed(communityId, feedType)`

**setTestGetFeedItemsFromFeed(communityId, feedType, pageParam, pageSize, sortParam, result)**

Register a `ConnectApi.FeedItemPage` object to be returned when `getFeedItemsFromFeed` is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

**API Version**

28.0–31.0

**Signature**

```java
public static Void setTestGetFeedItemsFromFeed(String communityId, ConnectApi.FeedType feedType, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedItemPage result)
```

**Parameters**

**communityId**

Type: `String`

ID for an Experience Cloud site, internal, or null.

**feedType**

Type: `ConnectApi.FeedType`

Type of feed. Valid values are `Company`, `DirectMessageModeration`, `DirectMessages`, `Home`, `Moderation`, and `PendingReview`.

**pageParam**

Type: `String`

Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in null, the first page is returned.
pageSize
Type: Integer
Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null, the default value CreatedDateDesc is used.

result
Type: ConnectApi.FeedItemPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
- getFeedItemsFromFeed(communityId, feedType, pageParam, pageSize, sortParam)

setTestGetFeedItemsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, result)
Register a ConnectApi.FeedItemPage object to be returned when getFeedItemsFromFeed is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

API Version
29.0–31.0

Signature
public static Void setTestGetFeedItemsFromFeed(String communityId, ConnectApi.FeedType feedType, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedItemPage result)
Parameters

**communityId**
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

**feedType**
Type: `ConnectApi.FeedType`
Type of feed. Valid values are `Company`, `DirectMessageModeration`, `DirectMessages`, `Home`, `Moderation`, and `PendingReview`.

**recentCommentCount**
Type: `Integer`
Maximum number of comments to return with each feed item. The default value is 3.

**density**
Type: `ConnectApi.FeedDensity`
Specify the amount of content in a feed.
- **AllUpdates**—Displays all update from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**
Type: `String`
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**
Type: `Integer`
Number of feed items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**
Type: `ConnectApi.FeedSortOrder`
Values are:
- **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- **Relevance**—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.

Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in `null`, the default value `CreatedDateDesc` is used.

**result**
Type: `ConnectApi.FeedItemPage`
Object containing test data.
Return Value
Type: Void

SEE ALSO:
getFeedItemsFromFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam)

**setTestGetFeedItemsFromFeed(communityId, feedType, subjectId, result)**

Register a `ConnectApi.FeedItemPage` object to be returned when `getFeedItemsFromFeed` is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

**API Version**
28.0–31.0

**Signature**
```java
public static Void setTestGetFeedItemsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, ConnectApi.FeedItemPage result)
```

**Parameters**
- **communityId**
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.
- **feedType**
  Type: `ConnectApi.FeedType`
  Type of feed. Valid values include every `ConnectApi.FeedType` except Company, DirectMessageModeration, DirectMessages, Filter, Home, Landing, Moderation, and PendingReview.
- **subjectId**
  Type: `String`
  If `feedType` is Record, `subjectId` can be any record ID, including a group ID. If `feedType` is Streams, `subjectId` must be a stream ID. If `feedType` is Topics, `subjectId` must be a topic ID. If `feedType` is UserProfile, `subjectId` can be any user ID. If the `feedType` is any other value, `subjectId` must be the ID of the context user or the alias `me`.
- **result**
  Type: `ConnectApi.FeedItemPage`
  Object containing test data.
Return Value
Type: Void

SEE ALSO:
getFeedItemsFromFeed(communityId, feedType, subjectId)

setTestGetFeedItemsFromFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam, result)
Register a ConnectApi.FeedItemPage object to be returned when getFeedItemsFromFeed is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

API Version
28.0–31.0

Signature
public static Void setTestGetFeedItemsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedItemPage result)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Type of feed. Valid values include every ConnectApi.FeedType except Company, DirectMessageModeration, DirectMessages, Filter, Home, Landing, Moderation, and PendingReview.

subjectId
Type: String
If feedType is Record, subjectId can be any record ID, including a group ID. If feedType is Streams, subjectId must be a stream ID. If feedType is Topics, subjectId must be a topic ID. If feedType is UserProfile, subjectId can be any user ID. If the feedType is any other value, subjectId must be the ID of the context user or the alias me.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.
sortParam
Type: ConnectApi.FeedSortOrder
Values are:
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null, the default value CreatedDateDesc is used.

result
Type: ConnectApi.FeedItemPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
- getFeedItemsFromFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam)

setTestGetFeedItemsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, result)

Register a ConnectApi.FeedItemPage object to be returned when getFeedItemsFromFeed is called with matching parameters in a test context. Use the get method with the same parameters or the code throws an exception.

API Version
29.0–31.0

Signature
public static Void setTestGetFeedItemsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedItemPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

**feedType**
Type: `ConnectApi.FeedType`
Type of feed. Valid values include every `ConnectApi.FeedType` except Company, DirectMessageModeration, DirectMessages, Filter, Home, Landing, Moderation, and PendingReview.

**subjectId**
Type: `String`
If `feedType` is Record, `subjectId` can be any record ID, including a group ID. If `feedType` is Streams, `subjectId` must be a stream ID. If `feedType` is Topics, `subjectId` must be a topic ID. If `feedType` is UserProfile, `subjectId` can be any user ID. If the `feedType` is any other value, `subjectId` must be the ID of the context user or the alias me.

**recentCommentCount**
Type: `Integer`
Maximum number of comments to return with each feed item. The default value is 3.

**density**
Type: `ConnectApi.FeedDensity`
Specify the amount of content in a feed.
- **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**
Type: `String`
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**
Type: `Integer`
Number of feed items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**
Type: `ConnectApi.FeedSortOrder`
Values are:
- **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for Home feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- **Relevance**—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in `null`, the default value `CreatedDateDesc` is used.

**result**
Type: `ConnectApi.FeedItemPage`
Object containing test data.

Return Value
Type: Void

SEE ALSO:
getFeedItemsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam)


setTestGetFeedItemsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, showInternalOnly, result)

Register a ConnectApi.FeedItemPage object to be returned when getFeedItemsFromFeed is called with matching parameters in a test context. Use the get feed method with the same parameters or the code throws an exception.

API Version
30.0–31.0

Signature
public static Void setTestGetFeedItemsFromFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, Boolean showInternalOnly, ConnectApi.FeedItemPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Type of feed. Valid values include every ConnectApi.FeedType except Company, DirectMessageModeration, DirectMessages, Filter, Home, Landing, Moderation, and PendingReview.

subjectId
Type: String
If feedType is Record, subjectId can be any record ID, including a group ID. If feedType is Streams, subjectId must be a stream ID. If feedType is Topics, subjectId must be a topic ID. If feedType is UserProfile, subjectId can be any user ID. If the feedType is any other value, subjectId must be the ID of the context user or the alias me.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed item. The default value is 3.
density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null, the default value CreatedDateDesc is used.

showInternalOnly
Type: Boolean
Specifies whether to show only feed items from internal (non-Experience Cloud site) users (true), or not (false). The default value is false.

result
Type: ConnectApi.FeedItemPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
- getFeedItemsFromFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, showInternalOnly)
setTestGetFeedItemsFromFilterFeed(communityId, subjectId, keyPrefix, result)

Register a ConnectApi.FeedItemPage object to be returned when the matching getFeedItemsFromFilterFeed method is called in a test context. Use the method with the same parameters or the code throws an exception.

API Version
28.0–31.0

Signature

public static Void setTestGetFeedItemsFromFilterFeed(String communityId, String subjectId, String keyPrefix, ConnectApi.FeedItemPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subjectId
Type: String
ID of the context user or the alias me.

keyPrefix
Type: String
A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

result
Type: ConnectApi.FeedItemPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
getFeedItemsFromFilterFeed(communityId, subjectId, keyPrefix)

setTestGetFeedItemsFromFilterFeed(communityId, subjectId, keyPrefix, pageParam, pageSize, sortParam, result)

Register a ConnectApi.FeedItemPage object to be returned when the matching getFeedItemsFromFilterFeed method is called in a test context. Use the method with the same parameters or the code throws an exception.

API Version
28.0–31.0
public static Void setTestGetFeedItemsFromFilterFeed(String communityId, String subjectId, String keyPrefix, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedItemPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subjectId
Type: String
ID of the context user or the alias me.

keyPrefix
Type: String
A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds. Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null, the default value CreatedDateDesc is used.

result
Type: ConnectApi.FeedItemPage
Object containing test data.
Return Value
Type: Void

SEE ALSO:
getFeedItemsFromFilterFeed(communityId, subjectId, keyPrefix, pageParam, pageSize, sortParam)

setTestGetFeedItemsFromFilterFeed(communityId, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, sortParam, result)
Register a ConnectApi.FeedItemPage object to be returned when the matching getFeedItemsFromFilterFeed method is called in a test context. Use the method with the same parameters or the code throws an exception.

API Version
29.0–31.0

Signature
public static Void setTestGetFeedItemsFromFilterFeed(String communityId, String subjectId, String keyPrefix, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedItemPage result)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subjectId
Type: String
ID of the context user or the alias me.

keyPrefix
Type: String
A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed item. The default value is 3.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.

• AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
• FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

**pageSize**
Type: Integer
Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

**sortParam**
Type: ConnectApi.FeedSortOrder
Values are:
• CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
• CreatedDateDesc—Sorts by most recent creation date.
• LastModifiedDateDesc—Sorts by most recent activity.
• MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
• Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds. Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null, the default value CreatedDateDesc is used.

**result**
Type: ConnectApi.FeedItemPage
Object containing test data.

**Return Value**
Type: Void

**SEE ALSO:**
getFeedItemsFromFilterFeed(communityId, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, sortParam)

setTestGetFeedItemsFromFilterFeedUpdatedSince(communityId, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, sortParam, updatedSince, result)
Register a ConnectApi.FeedItemPage object to be returned when the getFeedItemsFromFilterFeedUpdatedSince method is called in a test context.

**API Version**
30.0–31.0
public static Void setTestGetFeedItemsFromFilterFeedUpdatedSince(String communityId, String subjectId, String keyPrefix, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String updatedSince, ConnectApi.FeedItemPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subjectId
Type: String
ID of the context user or the alias me.

keyPrefix
Type: String
A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed item. The default value is 3.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.

Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds. Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null, the default value CreatedDateDesc is used.

updatedSince
Type: String
Opaque token containing information about the last modified date of the feed. Do not construct this token. To retrieve this token, call getFeedItemsFromFilterFeed and take the value from the updatesToken property of the ConnectApi.FeedItemPage response body.

result
Type: ConnectApi.FeedItemPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
getFeedItemsFromFilterFeedUpdatedSince(communityId, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, updatedSince)

setTestGetFeedItemsUpdatedSince(communityId, feedType, recentCommentCount, density, pageParam, pageSize, updatedSince, ConnectApi.FeedItemPage, results)
Register a ConnectApi.FeedItemPage object to be returned when getFeedItemsUpdatedSince is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

API Version
30.0–31.0

Signature
public static Void setTestGetFeedItemsUpdatedSince(String communityId, ConnectApi.FeedType feedType, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, String updatedSince, ConnectApi.FeedItemPage results)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Type of feed. Valid values are Company, DirectMessageModeration, DirectMessages, Home, Moderation, and PendingReview.

**recentCommentCount**
Type: Integer
Maximum number of comments to return with each feed item. The default value is 3.

**density**
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

**pageSize**
Type: Integer
Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

**updatedSince**
Type: String
An opaque token containing information about the last modified date of the feed. Do not construct this token. Retrieve this token from the updatesToken property of the ConnectApi.FeedItemPage response body.

**result**
Type: ConnectApi.FeedItemPage
Object containing test data.

**Return Value**
Type: Void

SEE ALSO:
- getFeedItemsUpdatedSince(communityId, feedType, recentCommentCount, density, pageParam, pageSize, updatedSince)

**setTestGetFeedItemsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince, result)**
Register a ConnectApi.FeedItemPage object to be returned when getFeedItemsUpdatedSince is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

**API Version**
30.0–31.0
public static Void setTestGetFeedItemsUpdatedSince(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, String updatedSince, ConnectApi.FeedItemPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
One of these values:
- Files
- Groups
- News
- People
- Record

subjectId
Type: String
If feedType is ConnectApi.Record, subjectId can be any record ID, including a group ID. Otherwise, it must be the context user or the alias me.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed item. The default value is 3.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

updatedSince
Type: String
An opaque token containing information about the last modified date of the feed. Do not construct this token. Retrieve this token from the `updatesToken` property of the `ConnectApi.FeedItemPage` response body.

```java
result
  Type: ConnectApi.FeedItemPage
  Object containing test data.
```

Return Value

Type: Void

SEE ALSO:

- `getFeedItemsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince)`
  *Apex Developer Guide: Testing ConnectApi Code*

```java
setTestGetFeedItemsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince, showInternalOnly, result)
```

Register a `ConnectApi.FeedItemPage` object to be returned when `getFeedItemsUpdatedSince` is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

API Version

30.0–31.0

Signature

```java
public static Void setTestGetFeedItemsUpdatedSince(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, String updatedSince, Boolean showInternalOnly, ConnectApi.FeedItemPage result)
```

Parameters

- `communityId`
  Type: String
  ID for an Experience Cloud site, `internal`, or `null`.

- `feedType`
  Type: `ConnectApi.FeedType`
  One of these values:
  - Files
  - Groups
  - News
  - People
  - Record
subjectId
Type: String
If `feedType` is `ConnectApi.Record`, `subjectId` can be any record ID, including a group ID. Otherwise, it must be the context user or the alias `me`.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed item. The default value is 3.

density
Type: `ConnectApi.FeedDensity`
Specify the amount of content in a feed.
- **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

pageSize
Type: Integer
Number of feed items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

updatedSince
Type: String
An opaque token containing information about the last modified date of the feed. Do not construct this token. Retrieve this token from the `updatesToken` property of the `ConnectApi.FeedItemPage` response body.

showInternalOnly
Type: Boolean
Specifies whether to show only feed items from internal (non-Experience Cloud site) users (`true`), or not (`false`). The default value is `false`.

result
Type: `ConnectApi.FeedItemPage`
Object containing test data.

Return Value
Type: Void

SEE ALSO:
- `getFeedItemsUpdatedSince(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, updatedSince, showInternalOnly)`
- *Apex Developer Guide: Testing ConnectApi Code*
**setTestSearchFeedItems**(communityId, q, result)

Register a test feed item page to be returned when `searchFeedItems(communityId, q)` is called during a test.

**API Version**
28.0–31.0

**Signature**

```java
public static Void searchFeedItems(String communityId, String q, ConnectApi.FeedItemPage result)
```

**Parameters**

- **communityId**
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.

- **q**
  Type: `String`
  Required and can't be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

- **result**
  Type: `ConnectApi.FeedItemPage`
  Object containing test data.

**Return Value**
Type: `Void`

**SEE ALSO:**
- `searchFeedItems(communityId, q)`

**setTestSearchFeedItems**(communityId, q, sortParam, result)

Register a test feed item page to be returned when `searchFeedItems(String, String, ConnectApi.FeedSortOrder)` is called during a test.

**API Version**
28.0–31.0

**Signature**

```java
public static Void setTestSearchFeedItems(String communityId, String q, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedItemPage result)
```
Parameters

**communityId**
- Type: String
- ID for an Experience Cloud site, internal, or null.

**q**
- Type: String
- Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

**sortParam**
- Type: ConnectApi.FeedSortOrder
- Values are:
  - CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
  - CreatedDateDesc—Sorts by most recent creation date.
  - LastModifiedDateDesc—Sorts by most recent activity.
  - MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
  - Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds. Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null, the default value CreatedDateDesc is used.

**result**
- Type: ConnectApi.FeedItemPage
- The feed item test page.

Return Value

Type: Void

SEE ALSO:
- searchFeedItems(communityId, q, sortParam)

setTestSearchFeedItems(communityId, q, pageParam, pageSize, result)

Register a test feed item page to be returned when searchFeedItems(String, String, String, Integer) is called during a test.

API Version

28.0–31.0
Signature

```java
public static Void setTestSearchFeedItems(String communityId, String q, String pageParam,
Integer pageSize, ConnectApi.FeedItemPage result)
```

Parameters

- `communityId`  
  Type: String  
  ID for an Experience Cloud site, internal, or `null`.
- `q`  
  Type: String  
  Required and can't be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See `Wildcards`.
- `pageParam`  
  Type: String  
  Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.
- `pageSize`  
  Type: Integer  
  Number of feed items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.
- `result`  
  Type: `ConnectApi.FeedItemPage`  
  The test feed item page.

Return Value

Type: Void

SEE ALSO:

- `searchFeedItems(communityId, q, pageParam, pageSize)`

```java
setTestSearchFeedItems(communityId, q, pageParam, pageSize, sortParam, result)
```

Register a test feed item page to be returned when `searchFeedItems(String, String, String, Integer, ConnectApi.FeedSortOrder)` is called during a test.

API Version

28.0–31.0

Signature

```java
public static Void setTestSearchFeedItems(String communityId, String q, String pageParam,
Integer pageSize, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedItemPage result)
```
Parameters

**communityId**
- Type: **String**
  - ID for an Experience Cloud site, internal, or **null**.

**q**
- Type: **String**
  - Required and can't be **null**. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See [Wildcards](#).

**pageParam**
- Type: **String**
  - Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in **null**, the first page is returned.

**pageSize**
- Type: **Integer**
  - Number of feed items per page. Valid values are from 1 through 100. If you pass in **null**, the default size is 25.

**sortParam**
- Type: **ConnectApi.FeedSortOrder**
  - Values are:
    - **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
    - **CreatedDateDesc**—Sorts by most recent creation date.
    - **LastModifiedDateDesc**—Sorts by most recent activity.
    - **MostViewed**—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
    - **Relevance**—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds. Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in **null**, the default value `CreatedDateDesc` is used.

**result**
- Type: **ConnectApi.FeedItemPage**
  - The test feed item page.

Return Value

Type: **Void**

**SEE ALSO:**
- `searchFeedItems(communityId, q, pageParam, pageSize, sortParam)`
- [Apex Developer Guide: Testing ConnectApi Code](#)
setTestSearchFeedItems(communityId, q, recentCommentCount, pageParam, pageSize, sortParam, result)

Register a test feed item page to be returned when searchFeedItems(communityId, q, recentCommentCount, pageParam, pageSize, sortParam) is called during a test.

API Version
29.0–31.0

Signature
public static Void setTestSearchFeedItems(String communityId, String q, Integer recentCommentCount, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, ConnectApi.FeedItemPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed item. The default value is 3.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder
Values are:
• CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
• CreatedDateDesc—Sorts by most recent creation date.
• LastModifiedDateDesc—Sorts by most recent activity.
• MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
• Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in \texttt{null}, the default value \texttt{CreatedDateDesc} is used.

\texttt{result}

Type: \texttt{ConnectApi.FeedItemPage}

The test feed item page.

Return Value

Type: \texttt{Void}

SEE ALSO:

searchFeedItems(communityId, q, recentCommentCount, pageParam, pageSize, sortParam)


\texttt{setTestSearchFeedItemsInFeed(communityId, feedType, q, result)}

Register a \texttt{ConnectApi.FeedItemPage} object to be returned when the matching \texttt{ConnectApi.searchFeedItemsInFeed} method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version

28.0–31.0

Signature

\texttt{public static Void setTestSearchFeedItemsInFeed(String communityId, ConnectApi.FeedType feedType, String q, ConnectApi.FeedItemPage result)}

Parameters

\texttt{communityId}

Type: \texttt{String}

ID for an Experience Cloud site, internal, or \texttt{null}.

\texttt{feedType}

Type: \texttt{ConnectApi.FeedType}

Type of feed. Valid values are Company, DirectMessageModeration, DirectMessages, Home, Moderation, and PendingReview.

\texttt{q}

Type: \texttt{String}

Required and can’t be \texttt{null}. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

\texttt{result}

Type: \texttt{ConnectApi.FeedItemPage}

Object containing test data.
setTestSearchFeedItemsInFeed(communityId, feedType, pageParam, pageSize, sortParam, q, result)

Register a ConnectApi.FeedItemPage object to be returned when the matching ConnectApi.searchFeedItemsInFeed method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
28.0–31.0

Signature

public static Void setTestSearchFeedItemsInFeed(String communityId, ConnectApi.FeedType feedType, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q, ConnectApi.FeedItemPage result)

Parameters

communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

feedType
  Type: ConnectApi.FeedType
  Type of feed. Valid values include every ConnectApi.FeedType except Company, DirectMessageModeration, Draft, DirectMessages, Filter, Home, Landing, Moderation, and PendingReview.

pageParam
  Type: String
  Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
  Type: Integer
  Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
  Type: ConnectApi.FeedSortOrder
  Values are:
  • CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
• CreatedDateDesc—Sorts by most recent creation date.
• LastModifiedDateDesc—Sorts by most recent activity.
• MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the
  ConnectApi.FeedFilter is UnansweredQuestions.
• Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null,
the default value CreatedDateDesc is used.

q
Type: String
Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including
wildcards. See Wildcards.

result
Type: ConnectApi.FeedItemPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
  searchFeedItemsInFeed(communityId, feedType, pageParam, pageSize, sortParam, q)

setTestSearchFeedItemsInFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, q, result)

Register a ConnectApi.FeedItemPage object to be returned when the matching
ConnectApi.searchFeedItemsInFeed method is called in a test context. Use the method with the same parameters or you
receive an exception.

API Version
29.0–31.0

Signature
public static Void setTestSearchFeedItemsInFeed(String communityId, ConnectApi.FeedType feedType, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q, ConnectApi.FeedItemPage result)

Parameters
  communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.
**feedType**  
Type: `ConnectApi.FeedType`  
Type of feed. Valid values include every `ConnectApi.FeedType` except Company, DirectMessageModeration, DirectMessages, Filter, Home, Landing, Moderation, and PendingReview.

**recentCommentCount**  
Type: `Integer`  
Maximum number of comments to return with each feed item. The default value is 3.

**density**  
Type: `ConnectApi.FeedDensity`  
Specify the amount of content in a feed.
- **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**  
Type: `String`  
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**  
Type: `Integer`  
Number of feed items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**  
Type: `ConnectApi.FeedSortOrder`  
Values are:
- **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for Home feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- **Relevance**—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds. Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in `null`, the default value `CreatedDateDesc` is used.

**q**  
Type: `String`  
Required and can't be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

**result**  
Type: `ConnectApi.FeedItemPage`  
Object containing test data.
Return Value
Type: Void

SEE ALSO:
searchFeedItemsInFeed(communityId, feedType, recentCommentCount, density, pageParam, pageSize, sortParam, q)


setTestSearchFeedItemsInFeed(communityId, feedType, subjectId, q, result)

Register a ConnectApi.FeedItemPage object to be returned when the matching ConnectApi.searchFeedItemsInFeed method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
28.0–31.0

Signature
public static Void setTestSearchFeedItemsInFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, String q, ConnectApi.FeedItemPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Type of feed. Valid values include every ConnectApi.FeedType except Company, DirectMessages, Filter, Landing, and Streams.

subjectId
Type: String
If feedType is Record, subjectId can be any record ID, including a group ID. If feedType is Streams, subjectId must be a stream ID. If feedType is Topics, subjectId must be a topic ID. If feedType is UserProfile, subjectId can be any user ID. If the feedType is any other value, subjectId must be the ID of the context user or the alias me.

q
Type: String
Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

result
Type: ConnectApi.FeedItemPage
Object containing test data.
Return Value
Type: Void

SEE ALSO:
searchFeedItemsInFeed(communityId, feedType, subjectId, q)


setTestSearchFeedItemsInFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam, q, result)

Register a ConnectApi.FeedItemPage object to be returned when the matching
ConnectApi.searchFeedItemsInFeed method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
28.0–31.0

Signature
public static Void setTestSearchFeedItemsInFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q, ConnectApi.FeedItemPage result)

Parameters
communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

feedType
  Type: ConnectApi.FeedType
  Type of feed. Valid values include every ConnectApi.FeedType except Company, DirectMessages, Filter, Landing, and Streams.

subjectId
  Type: String
  If feedType is Record, subjectId can be any record ID, including a group ID. If feedType is Streams, subjectId must be a stream ID. If feedType is Topics, subjectId must be a topic ID. If feedType is UserProfile, subjectId can be any user ID. If the feedType is any other value, subjectId must be the ID of the context user or the alias me.

pageParam
  Type: String
  Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
  Type: Integer
  Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.
sortParam
Type: ConnectApi.FeedSortOrder
Values are:

- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null, the default value CreatedDateDesc is used.

q
Type: String
Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

result
Type: ConnectApi.FeedItemPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
- searchFeedItemsInFeed(communityId, feedType, subjectId, pageParam, pageSize, sortParam, q)

setTestSearchFeedItemsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, result)

Register a ConnectApi.FeedItemPage object to be returned when the matching ConnectApi.searchFeedItemsInFeed method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
29.0–31.0

Signature
public static Void setTestSearchFeedItemsInFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q, ConnectApi.FeedItemPage result)
Parameters

**communityId**
- Type: String
- ID for an Experience Cloud site, internal, or null.

**feedType**
- Type: `ConnectApi.FeedType`
- Type of feed. Valid values include every `ConnectApi.FeedType` except Company, DirectMessages, Filter, Landing, and Streams.

**subjectId**
- Type: String
- If `feedType` is Record, `subjectId` can be any record ID, including a group ID. If `feedType` is Streams, `subjectId` must be a stream ID. If `feedType` is Topics, `subjectId` must be a topic ID. If `feedType` is UserProfile, `subjectId` can be any user ID. If the `feedType` is any other value, `subjectId` must be the ID of the context user or the alias me.

**recentCommentCount**
- Type: Integer
- Maximum number of comments to return with each feed item. The default value is 3.

**density**
- Type: `ConnectApi.FeedDensity`
- Specify the amount of content in a feed.
  - **AllUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
  - **FewerUpdates**—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

**pageParam**
- Type: String
- Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**
- Type: Integer
- Number of feed items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**
- Type: `ConnectApi.FeedSortOrder`
- Values are:
  - **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
  - **CreatedDateDesc**—Sorts by most recent creation date.
  - **LastModifiedDateDesc**—Sorts by most recent activity.
  - **MostViewed**—Sorts by most viewed content. This sort order is available only for Home feeds when the `ConnectApi.FeedFilter` is UnansweredQuestions.
  - **Relevance**—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.
Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in `null`, the default value `CreatedDateDesc` is used.

q
Type: `String`
Required and can’t be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See `Wildcards`.

result
Type: `ConnectApi.FeedItemPage`
Object containing test data.

Return Value
Type: `Void`

SEE ALSO:
`searchFeedItemsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q)`

```
setTestSearchFeedItemsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, showInternalOnly, result)
```

Register a `ConnectApi.FeedItemPage` object to be returned when the matching `ConnectApi.searchFeedItemsInFeed` method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
29.0–31.0

Signature
```
public static Void setTestSearchFeedItemsInFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q, Boolean showInternalOnly, ConnectApi.FeedItemPage result)
```

Parameters

- `communityId`
  Type: `String`
  ID for an Experience Cloud site, `internal`, or `null`.

- `feedType`
  Type: `ConnectApi.FeedType`
  Type of feed. Valid values include every `ConnectApi.FeedType` except `Company`, `DirectMessages`, `Filter`, `Landing`, and `Streams`.

- `subjectId`
  Type: `String`
  Subject ID for the search feed item.

- `recentCommentCount`
  Type: `Integer`
  Maximum number of recent comments to return.

- `density`
  Type: `ConnectApi.FeedDensity`
  Feed item density. Valid values include `All`, `Medium`, `Low`, and `High`.

- `pageParam`
  Type: `String`
  Custom query parameter.

- `pageSize`
  Type: `Integer`
  Number of items to return per page.

- `sortParam`
  Type: `ConnectApi.FeedSortOrder`
  Feed item sort order.

- `q`
  Type: `String`
  Search string. See `Wildcards`.

- `showInternalOnly`
  Type: `Boolean`
  Whether to show internal feeds.

- `result`
  Type: `ConnectApi.FeedItemPage`
  Object containing test data.
subjectId
Type: String

If feedType is Record, subjectId can be any record ID, including a group ID. If feedType is Streams, subjectId
must be a stream ID. If feedType is Topics, subjectId must be a topic ID. If feedType is UserProfile, subjectId
can be any user ID. If the feedType is any other value, subjectId must be the ID of the context user or the alias me.

recentCommentCount
Type: Integer

Maximum number of comments to return with each feed item. The default value is 3.

density
Type: ConnectApi.FeedDensity

Specify the amount of content in a feed.

- AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
- FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.

pageParam
Type: String

Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken
or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer

Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.FeedSortOrder

Values are:

- CreatedDateAsc—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration,
  Draft, Moderation, and PendingReview feeds.
- CreatedDateDesc—Sorts by most recent creation date.
- LastModifiedDateDesc—Sorts by most recent activity.
- MostViewed—Sorts by most viewed content. This sort order is available only for Home feeds when the
  ConnectApi.FeedFilter is UnansweredQuestions.
- Relevance—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null,
the default value CreatedDateDesc is used.

q
Type: String

Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

showInternalOnly
Type: Boolean
Specifies whether to show only feed items from internal (non-Experience Cloud site) users (true), or not (false). The default value is false.

result
Type: ConnectApi.FeedItemPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
searchFeedItemsInFeed(communityId, feedType, subjectId, recentCommentCount, density, pageParam, pageSize, sortParam, q, showInternalOnly)


setTestSearchFeedItemsInFilterFeed(communityId, subjectId, keyPrefix, q, result)

Register a ConnectApi.FeedItemPage object to be returned when the matching ConnectApi.searchFeedItemsInFilterFeed method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
28.0–31.0

Signature
public static Void setTestSearchFeedItemsInFilterFeed(String communityId, String subjectId, String keyPrefix, String q, ConnectApi.FeedItemPage result)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

subjectId
Type: String
ID of the context user or the alias me.

keyPrefix
Type: String
A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

q
Type: String
Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

`result`  
Type: `ConnectApi.FeedItemPage`  
Specify the test feed item page.

**Return Value**  
Type: Void

SEE ALSO:  
`searchFeedItemsInFilterFeed(communityId, subjectId, keyPrefix, q)`  
*Apex Developer Guide: Testing ConnectApi Code*

```plaintext
setTestSearchFeedItemsInFilterFeed(communityId, feedType, subjectId, keyPrefix, pageParam, pageSize, sortParam, q, result)
```

Register a `ConnectApi.FeedItemPage` object to be returned when the matching `ConnectApi.searchFeedItemsInFilterFeed` method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**  
28.0–31.0

**Signature**  
```plaintext
public static Void setTestSearchFeedItemsInFilterFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, String keyPrefix, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q, ConnectApi.FeedItemPage result)
```

**Parameters**

- **communityId**  
  Type: `String`  
  ID for an Experience Cloud site, internal, or null.

- **feedType**  
  Type: `ConnectApi.FeedType`  
  Type of feed. Valid values include every `ConnectApi.FeedType` except Company, DirectMessageModeration, DirectMessages, Filter, Home, Landing, Moderation, and PendingReview.

- **subjectId**  
  Type: `String`  
  ID of the context user or the alias me.

- **keyPrefix**  
  Type: `String`
A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

**pageParam**
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

**pageSize**
Type: Integer
Number of feed items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

**sortParam**
Type: ConnectApi.FeedSortOrder
Values are:
- **CreateDateAsc**—Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.
- **CreateDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.
- **Relevance**—Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.

Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in null, the default value CreatedDateDesc is used.

**q**
Type: String
Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

**result**
Type: ConnectApi.FeedItemPage
Specify the test feed item page.

**Return Value**
Type: Void

SEE ALSO:
- `searchFeedItemsInFilterFeed(communityId, subjectId, keyPrefix, pageParam, pageSize, sortParam, q)`
- *Apex Developer Guide: Testing ConnectApi Code*
setTestSearchFeedItemsInFilterFeed(communityId, feedType, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, sortParam, q, result)

Register a ConnectApi.FeedItemPage object to be returned when the matching ConnectApi.searchFeedItemsInFilterFeed method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
29.0–31.0

Signature
public static Void setTestSearchFeedItemsInFilterFeed(String communityId, ConnectApi.FeedType feedType, String subjectId, String keyPrefix, Integer recentCommentCount, ConnectApi.FeedDensity density, String pageParam, Integer pageSize, ConnectApi.FeedSortOrder sortParam, String q, ConnectApi.FeedItemPage result)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedType
Type: ConnectApi.FeedType
Type of feed. Valid values include every ConnectApi.FeedType except Company, DirectMessageModeration, DirectMessages, Filter, Home, Landing, Moderation, and PendingReview.

subjectId
Type: String
ID of the context user or the alias me.

keyPrefix
Type: String
A key prefix that specifies record type. A key prefix is the first three characters in the object ID, which specifies the object type. For example, User objects have a prefix of 00S and Group objects have a prefix of 0F9.

recentCommentCount
Type: Integer
Maximum number of comments to return with each feed item. The default value is 3.

density
Type: ConnectApi.FeedDensity
Specify the amount of content in a feed.
• AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.
• FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.
**pageParam**
Type: **String**
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**
Type: **Integer**
Number of feed items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**
Type: **ConnectApi.FeedSortOrder**
Values are:
- **CreatedDateAsc**—Sorts by oldest creation date. This sort order is available only for `DirectMessageModeration`, `Draft`, `Moderation`, and `PendingReview` feeds.
- **CreatedDateDesc**—Sorts by most recent creation date.
- **LastModifiedDateDesc**—Sorts by most recent activity.
- **MostViewed**—Sorts by most viewed content. This sort order is available only for `Home` feeds when the `ConnectApi.FeedFilter` is `UnansweredQuestions`.
- **Relevance**—Sorts by most relevant content. This sort order is available only for `Company`, `Home`, and `Topics` feeds.

Sorts the returned feed by the most recently created feed item, or by the most recently modified feed item. If you pass in `null`, the default value `CreatedDateDesc` is used.

**q**
Type: **String**
Required and can’t be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

**result**
Type: **ConnectApi.FeedItemPage**
Specify the test feed item page.

**Return Value**
Type: Void

SEE ALSO:
- `searchFeedItemsInFilterFeed(communityId, subjectId, keyPrefix, recentCommentCount, density, pageParam, pageSize, sortParam, q)`
  *Apex Developer Guide: Testing ConnectApi Code*

**ChatterGroups Class**
Information about groups, such as the group’s members, photo, and the groups the specified user is a member of. Add members to a group, remove members, and change the group photo.
Namespace

ConnectApi

ChatterGroups Methods

The following are methods for ChatterGroups. All methods are static.

IN THIS SECTION:

- `addMember(communityId, groupId, userId)`: Add a user to a group as a standard member.
- `addMemberWithRole(communityId, groupId, userId, role)`: Add a user with a role to a group.
- `addRecord(communityId, groupId, recordId)`: Associate a record with a group.
- `createGroup(communityId, groupInput)`: Create a group.
- `deleteBannerPhoto(communityId, groupId)`: Delete the group banner photo.
- `deleteGroup(communityId, groupId)`: Delete a group.
- `deleteMember(communityId, membershipId)`: Remove a member from a group.
- `deletePhoto(communityId, groupId)`: Delete the group photo.
- `getAnnouncements(communityId, groupId)`: Get the first page of announcements in a group.
- `getAnnouncements(communityId, groupId, pageParam, pageSize)`: Get a page of announcements in a group.
- `getBannerPhoto(communityId, groupId)`: Get the group banner photo.
- `getGroup(communityId, groupId)`: Get information about a group.
- `getGroupBatch(communityId, groupIds)`: Get information about a list of groups.
- `getGroupMembershipRequest(communityId, requestId)`: Get information about a request to join a private group.
- `getGroupMembershipRequests(communityId, groupId)`: Get information about every request to join a private group.
- `getGroupMembershipRequests(communityId, groupId, status)`: Get information about every request to join a private group that has a specified status.
getGroups(communityId)
Get the first page of groups.

groups(communityId, pageParam, pageSize)
Get a page of groups.

groups(communityId, pageParam, pageSize, archiveStatus)
Get a page of groups with an archive status.

getMember(communityId, membershipId)
Get information about a group member.

getMembers(communityId, groupId)
Get the first page of information about the members of a group.

groups(communityId, groupId, pageParam, pageSize)
Get a page of information about the members of a group.

getMembershipBatch(communityId, membershipIds)
Get information about a list of group memberships.

getMyChatterSettings(communityId, groupId)
Get the context user’s Chatter settings for a group.

groups(communityId, groupId)
Get the photo for a group.

getRecord(communityId, groupRecordId)
Get a record associated with a group.

getRecords(communityId, groupId)
Get the first page of records associated with a group.

groups(communityId, groupId, pageParam, pageSize)
Get a page of records associated with a group.

inviteUsers(groupId, invite)
Invite internal and external users to join a group.

postAnnouncement(communityId, groupId, announcement)
Post an announcement to a group.

removeRecord(communityId, groupRecordId)
Remove the association of a record with a group.

requestGroupMembership(communityId, groupId)
Request membership in a private group.

searchGroups(communityId, q)
Get the first page of groups that match the search criteria.

searchGroups(communityId, q, pageParam, pageSize)
Get a page of groups that match the search criteria.

searchGroups(communityId, q, archiveStatus, pageParam, pageSize)
Get a page of groups with the archive status that match the search criteria.

setBannerPhoto(communityId, groupId, fileId, versionNumber)
Set an uploaded file as the group banner photo.
setBannerPhoto(communityId, groupId, fileUpload)
Set a file that hasn’t been uploaded as the group banner photo.

setBannerPhotoWithAttributes(communityId, groupId, bannerPhoto)
Set and crop an uploaded file as the group banner photo.

setBannerPhotoWithAttributes(communityId, groupId, bannerPhoto, fileUpload)
Set and crop a file that hasn’t been uploaded as the group banner photo.

setPhoto(communityId, groupId, fileId, versionNumber)
Set an uploaded file as the group photo.

setPhoto(communityId, groupId, fileUpload)
Set a file that hasn’t been uploaded as the group photo.

setPhotoWithAttributes(communityId, groupId, photo)
Set and crop an uploaded file as the group photo.

setPhotoWithAttributes(communityId, groupId, photo, fileUpload)
Set and crop a file that hasn’t been uploaded as the group photo.

updateGroup(communityId, groupId, groupInput)
Update the settings of a group.

updateGroupMember(communityId, membershipId, role)
Update the role of a group member.

updateMyChatterSettings(communityId, groupId, emailFrequency)
Update the context user’s email frequency for a group.

updateRequestStatus(communityId, requestId, status)
Update a request to join a private group.

updateRequestStatus(communityId, requestId, status, responseMessage)
Update a request to join a private group and optionally provide a message when the request is denied.

addMember(communityId, groupId, userId)
Add a user to a group as a standard member.

API Version
28.0

Requires Chatter
Yes

Signature
public static ConnectApi.GroupMember addMember(String communityId, String groupId, String userId)
Parameters

- **communityId**
  Type: String
  ID for an Experience Cloud site, internal, or null.

- **groupId**
  Type: String
  ID for a group.

- **userId**
  Type: String
  ID for a user.

Return Value

Type: ConnectApi.GroupMember

Usage

To execute this method, the context user must be the group owner or moderator.

```
addMemberWithRole(communityId, groupId, userId, role)
```

Add a user with a role to a group.

API Version

29.0

Requires Chatter

Yes

Signature

```
public static ConnectApi.GroupMember addMemberWithRole(String communityId, String groupId, String userId, ConnectApi.GroupMembershipType role)
```

Parameters

- **communityId**
  Type: String
  ID for an Experience Cloud site, internal, or null.

- **groupId**
  Type: String
  ID for a group.

- **userId**
  Type: String
ID for a user.

**role**
Type: `ConnectApi.GroupMembershipType`
The group membership type. One of these values:
- GroupManager
- StandardMember

**Return Value**
Type: `ConnectApi.GroupMember`

**Usage**
To execute this method, the context user must be the group owner or moderator.

**addRecord(communityId, groupId, recordId)**
Associate a record with a group.

**API Version**
34.0

**Requires Chatter**
Yes

**Signature**
```
public static ConnectApi.GroupRecord addRecord(String communityId, String groupId, String recordId)
```

**Parameters**
- **communityId**
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.
- **groupId**
  Type: `String`
  ID of the group with which to associate the record.
- **recordId**
  Type: `String`
  ID of the record to associate with the group.

**Return Value**
Type: `ConnectApi.GroupRecord`
createGroup(communityId, groupInput)
Create a group.

API Version
29.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterGroupDetail createGroup(String communityId, ConnectApi.ChatterGroupInput groupInput)

Parameters

communityId
Type: String,
ID for an Experience Cloud site, internal, or null.

groupInput
Type: ConnectApi.ChatterGroupInput
The properties of the group.

Return Value
Type: ConnectApi.ChatterGroupDetail

deleteBannerPhoto(communityId, groupId)
Delete the group banner photo.

API Version
36.0

Requires Chatter
Yes

Signature
public static Void deleteBannerPhoto(String communityId, String groupId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

groupId
  Type: String
  ID of the group.

Return Value
Type: Void

Usage
This method is successful only when the context user is the group manager or owner, or has Modify All Data permission.

deleteGroup(communityId, groupId)
Delete a group.

API Version
29.0

Requires Chatter
Yes

Signature
public static Void deleteGroup(String communityId, String groupId)

Parameters
communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

groupId
  Type: String
  ID for a group.

Return Value
Type: Void

deleteMember(communityId, memberId)
Remove a member from a group.

API Version
28.0
Requires Chatter
Yes

Signature
public static Void deleteMember(String communityId, String membershipId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

membershipId
Type: String
ID for a membership.

Return Value
Type: Void

Usage
This method is successful only when the context user is the group manager or owner, or has Modify All Data permission.

deletePhoto(communityId, groupId)
Delete the group photo.

API Version
28.0

Requires Chatter
Yes

Signature
public static Void deletePhoto(String communityId, String groupId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

groupId
Type: String
ID for a group.
Return Value
Type: Void

Usage
This method is only successful when the context user is the group manager or owner, or has Modify All Data permission.

getAnnouncements(communityId, groupId)
Get the first page of announcements in a group.

API Version
31.0

Requires Chatter
Yes

Signature
public static ConnectApi.AnnouncementPage getAnnouncements(String communityId, String groupId)

Parameters
communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

groupId
  Type: String
  ID for a group.

Return Value
Type: ConnectApi.AnnouncementPage

Usage
To post an announcement, get information about an announcement, update the expiration date of an announcement, or delete an announcement, use the methods of the ConnectApi.Announcements class.

getAnnouncements(communityId, groupId, pageParam, pageSize)
Get a page of announcements in a group.

API Version
31.0
Requires Chatter
Yes

Signature

```java
public static ConnectApi.AnnouncementPage getAnnouncements(String communityId, String groupId, Integer pageParam, Integer pageSize)
```

Parameters

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **groupId**
  - Type: `String`
  - ID for a group.

- **pageParam**
  - Type: `Integer`
  - Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

- **pageSize**
  - Type: `Integer`
  - Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

Return Value

Type: `ConnectApi.AnnouncementPage`

Usage

To post an announcement, get information about an announcement, update the expiration date of an announcement, or delete an announcement, use the methods of the `ConnectApi.Announcements` class.

`getBannerPhoto(communityId, groupId)`

Get the group banner photo.

API Version

36.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.BannerPhoto getBannerPhoto(String communityId, String groupId)
```
Parameters

**communityId**
Type: String
ID for an Experience Cloud site, internal, or null.

**groupId**
Type: String
The ID of the group.

Return Value
Type: `ConnectApi.BannerPhoto`

**getGroup(communityId, groupId)**
Get information about a group.

API Version
28.0

Available to Guest Users
31.0

Requires Chatter
Yes

Signature

```
public static ConnectApi.ChatterGroupDetail getGroup(String communityId, String groupId)
```

Parameters

**communityId**
Type: String
ID for an Experience Cloud site, internal, or null.

**groupId**
Type: String
ID for a group.

Return Value
Type: `ConnectApi.ChatterGroupDetail`

**getGroupBatch(communityId, groupIds)**
Get information about a list of groups.
API Version
31.0

Requires Chatter
Yes

Signature
public static ConnectApi.BatchResult[] getGroupBatch(String communityId, List<String> groupIds)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

groupId
Type: List<String>
A list of up to 500 group IDs.

Return Value
Type: ConnectApi.BatchResult[]
The ConnectApi.BatchResult.getResult() method returns a ConnectApi.ChatterGroup object and errors embedded in the results for groups that didn’t load.

Example
```
// Create a list of groups.
ConnectApi.ChatterGroupPage groupPage = ConnectApi.ChatterGroups.getGroups(null);

// Create a list of group IDs.
List<String> groupIds = new List<String>();
for (ConnectApi.ChatterGroup aGroup : groupPage.groups){
    groupIds.add(aGroup.id);
}

// Get info about all the groups in the list.
ConnectApi.BatchResult[] batchResults = ConnectApi.ChatterGroups.getGroupBatch(null, groupIds);

for (ConnectApi.BatchResult batchResult : batchResults) {
    if (batchResult.isSuccess()) {
        // Operation was successful.
        ConnectApi.ChatterGroup aGroup;
        if (batchResult.getResult() instanceof ConnectApi.ChatterGroup) {
            aGroup = (ConnectApi.ChatterGroup) batchResult.getResult();
        }
    }
```
System.debug('SUCCESS');
System.debug(aGroup.memberCount);
}
else {
    // Operation failed. Print errors.
    System.debug('FAILURE');
    System.debug(batchResult.getErrorMessage());
}

SEE ALSO:
getMembershipBatch(communityId, membershipIds)

getGroupMembershipRequest(communityId, requestId)
Get information about a request to join a private group.

API Version
28.0

Requires Chatter
Yes

Signature
public static ConnectApi.GroupMembershipRequest getGroupMembershipRequest(String communityId, String requestId)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.
requestId
Type: String
The ID of a request to join a private group.

Return Value
Type: ConnectApi.GroupMembershipRequest

Usage
This method is successful only when the context user is the group manager or owner, or has Modify All Data permission.
**getGroupMembershipRequests(communityId, groupId)**

Get information about every request to join a private group.

API Version

28.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.GroupMembershipRequests getGroupMembershipRequests(String communityId, String groupId)
```

Parameters

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or null.

- **groupId**
  - Type: `String`
  - ID for a group.

Return Value

Type: `ConnectApi.GroupMembershipRequests`

Usage

This method is successful only when the context user is the group manager or owner, or has Modify All Data permission.

**getGroupMembershipRequests(communityId, groupId, status)**

Get information about every request to join a private group that has a specified status.

API Version

28.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.GroupMembershipRequests getGroupMembershipRequests(String communityId, String groupId, ConnectApi.GroupMembershipRequestStatus status)
```
Parameters

`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

`groupId`
Type: `String`
ID for a group.

`status`
Type: `ConnectApi.GroupMembershipRequestStatus`
  `status`—Status of a request to join a private group.
  - Accepted
  - Declined
  - Pending

Return Value
Type: `ConnectApi.GroupMembershipRequests`

Usage
This method is successful only when the context user is the group manager or owner, or has Modify All Data permission.

`getGroups(communityId)`
Get the first page of groups.

API Version
28.0

Available to Guest Users
31.0

Requires Chatter
Yes

Signature
`public static ConnectApi.ChatterGroupPage getGroups(String communityId)`

Parameters

`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or `null`. 
Return Value
Type: `ConnectApi.ChatterGroupPage`

`getGroups(communityId, pageParam, pageSize)`
Get a page of groups.

API Version
28.0

Available to Guest Users
31.0

Requires Chatter
Yes

Signature
`public static ConnectApi.ChatterGroupPage getGroups(String communityId, Integer pageParam, Integer pageSize)`

Parameters
- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.
- `pageParam`
  Type: `Integer`
  Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.
- `pageSize`
  Type: `Integer`
  Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

Return Value
Type: `ConnectApi.ChatterGroupPage`

`getGroups(communityId, pageParam, pageSize, archiveStatus)`
Get a page of groups with an archive status.

API Version
29.0
Available to Guest Users
31.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterGroupPage getGroups(String communityId, Integer pageParam, Integer pageSize, ConnectApi.GroupArchiveStatus archiveStatus)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

archiveStatus
Type: ConnectApi.GroupArchiveStatus
Archive status of groups.

• All—All groups, including groups that are archived and groups that aren’t archived.
• Archived—Groups that are archived.
• NotArchived—Groups that aren’t archived.
If you pass in null, the default value is All.

Return Value
Type: ConnectApi.ChatterGroupPage

getMember(communityId, membershipId)
Get information about a group member.

API Version
28.0

Requires Chatter
Yes
Signature

```java
public static ConnectApi.GroupMember getMember(String communityId, String membershipId)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- `membershipId`
  - Type: `String`
  - ID for a membership.

Return Value

Type: `ConnectApi.GroupMember`

```java
getMembers(communityId, groupId)
```

Get the first page of information about the members of a group.

API Version

28.0

Available to Guest Users

36.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.GroupMemberPage getMembers(String communityId, String groupId)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- `groupId`
  - Type: `String`
  - ID for a group.

Return Value

Type: `ConnectApi.GroupMemberPage`
**getMembers(communityId, groupId, pageParam, pageSize)**
Get a page of information about the members of a group.

**API Version**
28.0

**Available to Guest Users**
36.0

**Requires Chatter**
Yes

**Signature**
```
public static ConnectApi.GroupMemberPage getMembers(String communityId, String groupId, Integer pageParam, Integer pageSize)
```

**Parameters**
- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.
- **groupId**
  - Type: String
  - ID for a group.
- **pageParam**
  - Type: Integer
  - Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.
- **pageSize**
  - Type: Integer
  - Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

**Return Value**
Type: ConnectApi.GroupMemberPage

**getMembershipBatch(communityId, membershipIds)**
Get information about a list of group memberships.

**API Version**
31.0
Requires Chatter
Yes

Signature

```java
public static ConnectApi.BatchResult[] getMembershipBatch(String communityId,
List<String> membershipIds)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- `membershipIds`
  - Type: `List<String>`
  - A list of up to 500 group membership IDs.

Return Value

- Type: `ConnectApi.BatchResult[]`

The `ConnectApi.BatchResult.getResult()` method returns a `ConnectApi.GroupMember` object and errors embedded in the results for group memberships that didn’t load.

Example

```java
// Get members of a group.
ConnectApi.GroupMemberPage membersPage = ConnectApi.ChatterGroups.getMembers(null,
'0F9D00000000oOT');

// Create a list of membership IDs.
List<String> membersList = new List<String>();
for (ConnectApi.GroupMember groupMember : membersPage.members){
    membersList.add(groupMember.id);
}

// Get info about all group memberships in the list.
ConnectApi.BatchResult[] batchResults = ConnectApi.ChatterGroups.getMembershipBatch(null,
membersList);

for (ConnectApi.BatchResult batchResult : batchResults) {
    if (batchResult.isSuccess()) {
        // Operation was successful.
        // Print the first name of each member.
        ConnectApi.GroupMember groupMember;
        if(batchResult.getResult() instanceof ConnectApi.GroupMember) {
            groupMember = (ConnectApi.GroupMember) batchResult.getResult();
        } System.debug('SUCCESS');
        System.debug(groupMember.user.firstName);
    }
else {
    // Operation failed. Print errors.
    System.debug('FAILURE');
    System.debug(batchResult.getErrorMessage());
}

SEE ALSO:

getGroupBatch(communityId, groupIds)

getMyChatterSettings(communityId, groupId)
Get the context user’s Chatter settings for a group.

API Version
28.0

Requires Chatter
Yes

Signature
public static ConnectApi.GroupChatterSettings getMyChatterSettings(String communityId, String groupId)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

groupId
Type: String
ID for a group.

Return Value
Type: ConnectApi.GroupChatterSettings

getPhoto(communityId, groupId)
Get the photo for a group.

API Version
28.0
Requires Chatter
Yes

Signature

public static ConnectApi.Photo getPhoto(String communityId, String groupId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

groupId
Type: String
ID for a group.

Return Value
Type: ConnectApi.Photo

getRecord(communityId, groupRecordId)
Get a record associated with a group.

API Version
34.0

Requires Chatter
Yes

Signature

public static ConnectApi.GroupRecord getRecord(String communityId, String groupRecordId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

groupRecordId
Type: String
ID of the group record.

Return Value
Type: ConnectApi.GroupRecord
**getRecords(communityId, groupId)**

Get the first page of records associated with a group.

**API Version**

33.0

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.GroupRecordPage getRecords(String communityId, String groupId)
```

**Parameters**

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.
- **groupId**
  - Type: String
  - ID for a group.

**Return Value**

Type: `ConnectApi.GroupRecordPage`

**getRecords(communityId, groupId, pageParam, pageSize)**

Get a page of records associated with a group.

**API Version**

33.0

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.GroupRecordPage getRecords(String communityId, String groupId, Integer pageParam, Integer pageSize)
```

**Parameters**

- **communityId**
  - Type: String
ID for an Experience Cloud site, internal, or null.

groupId
Type: String
ID for a group.

groupId
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

Return Value
Type: ConnectApi.GroupRecordPage

inviteUsers(groupId, invite)
Invite internal and external users to join a group.

API Version
39.0

Requires Chatter
Yes

Signature
public static ConnectApi.Invitations inviteUsers(String groupId, ConnectApi.InviteInput invite)

Parameters

groupId
Type: String
ID of the group.

invite
Type: ConnectApi.InviteInput
A ConnectApi.InviteInput body.

Return Value
Type: ConnectApi.Invitations
postAnnouncement(communityId, groupId, announcement)

Post an announcement to a group.

API Version
31.0

Requires Chatter
Yes

Signature
public static ConnectApi.Announcement postAnnouncement(String communityId, String groupId, ConnectApi.AnnouncementInput announcement)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

groupId
Type: String
ID for a group.

announcement
Type: ConnectApi.AnnouncementInput
A ConnectApi.AnnouncementInput object.

Return Value
Type: ConnectApi.Announcement

Usage
Use an announcement to highlight information. Users can discuss, like, and post comments on announcements. Deleting the feed post deletes the announcement.

To post an announcement, get information about an announcement, update the expiration date of an announcement, or delete an announcement, use the methods of the ConnectApi.Announcements class.

removeRecord(communityId, groupRecordId)

Remove the association of a record with a group.

API Version
34.0
Requires Chatter
Yes

Signature

\[
\text{public static Void removeRecord(} \text{String communityId, String groupRecordId)}
\]

Parameters

\[
\text{communityId}
\]
Type: \text{String}
ID for an Experience Cloud site, internal, or \text{null}.

\[
\text{groupRecordId}
\]
Type: \text{String}
ID of the group record.

Return Value
Type: \text{Void}

\[
\text{requestGroupMembership(} \text{String communityId, String groupId)}
\]
Request membership in a private group.

API Version
28.0

Requires Chatter
Yes

Signature

\[
\text{public static ConnectApi.GroupMembershipRequest requestGroupMembership(} \text{String communityId, String groupId)}
\]

Parameters

\[
\text{communityId}
\]
Type: \text{String}
ID for an Experience Cloud site, internal, or \text{null}.

\[
\text{groupId}
\]
Type: \text{String}
ID for a group.
Return Value
Type: ConnectApi.GroupMembershipRequest

Sample: Requesting to Join a Private Group
This sample code calls ConnectApi.ChatterGroups.requestGroupMembership to request to join a private group.

```java
String communityId = null;
ID groupId = '0F9x00000000hAZ';

ConnectApi.GroupMembershipRequest membershipRequest =
ConnectApi.ChatterGroups.requestGroupMembership(communityId, groupId);
```

`searchGroups(communityId, q)`
Get the first page of groups that match the search criteria.

API Version
28.0

Available to Guest Users
31.0

Requires Chatter
Yes

Signature
`public static ConnectApi.ChatterGroupPage searchGroups(String communityId, String q)`

Parameters
`communityId`
Type: String
ID for an Experience Cloud site, internal, or null.

`q`
Type: String
Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards. Can be specified as null.

Return Value
Type: ConnectApi.ChatterGroupPage
Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
  setTestSearchGroups(communityId, q, result)

searchGroups(communityId, q, pageParam, pageSize)
Get a page of groups that match the search criteria.

API Version
28.0

Available to Guest Users
31.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterGroupPage searchGroups(String communityId, String q, Integer pageParam, Integer pageSize)

Parameters
  communityId
    Type: String
    ID for an Experience Cloud site, internal, or null.
  q
    Type: String
    Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards. Can be specified as null.
  pageParam
    Type: Integer
    Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.
  pageSize
    Type: Integer
    Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.
Return Value
Type: `ConnectApi.ChatterGroupPage`

Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestSearchGroups(communityId, q, pageParam, pageSize, result)`  
  *Apex Developer Guide: Testing ConnectApi Code*

`searchGroups(communityId, q, archiveStatus, pageParam, pageSize)`
Get a page of groups with the archive status that match the search criteria.

API Version
29.0

Available to Guest Users
31.0

Requires Chatter
Yes

Signature
```java
public static ConnectApi.ChatterGroupPage searchGroups(String communityId, String q, ConnectApi.GroupArchiveStatus archiveStatus, Integer pageParam, Integer pageSize)
```

Parameters
- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.
- `q`
  Type: `String`
  Specifies the string to search. The search string must contain at least two characters, not including wildcards. See `Wildcards`. Can be specified as `null`.
- `archiveStatus`
  Type: `ConnectApi.GroupArchiveStatus`
  Archive status of groups.
  - **All**—All groups, including groups that are archived and groups that aren’t archived.
  - **Archived**—Groups that are archived.
NotArchived—Groups that aren’t archived.

**pageParam**
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

**pageSize**
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**Return Value**
Type: `ConnectApi.ChatterGroupPage`

**Usage**
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestSearchGroups(communityId, q, archiveStatus, pageParam, pageSize, result)`
  *Apex Developer Guide: Testing ConnectApi Code*

### `setBannerPhoto(communityId, groupId, fileId, versionNumber)`
Set an uploaded file as the group banner photo.

**API Version**
36.0

**Requires Chatter**
Yes

**Signature**
```
public static ConnectApi.BannerPhoto setBannerPhoto(String communityId, String groupId, String fileId, Integer versionNumber)
```

**Parameters**
- **communityId**
  Type: String
  ID for an Experience Cloud site, internal, or `null`.
- **groupId**
  Type: String
  The ID of the group.
**fileId**
Type: *String*
The ID of the already uploaded file. The key prefix must be 069, and the image must be smaller than 8 MB.

**versionNumber**
Type: *Integer*
Version number of the existing file. Specify either an existing version number or, to get the latest version, specify `null`.

**Return Value**
Type: *ConnectApi.BannerPhoto*

**Usage**
This method is successful only when the context user is the group manager or owner, or has Modify All Data permission. Photos are processed asynchronously and might not be visible right away.

**setBannerPhoto(communityId, groupId, fileUpload)**
Set a file that hasn’t been uploaded as the group banner photo.

**API Version**
36.0

**Requires Chatter**
Yes

**Signature**
```java
public static ConnectApi.BannerPhoto setBannerPhoto(String communityId, String groupId, ConnectApi.BinaryInput fileUpload)
```

**Parameters**

**communityId**
Type: *String*
ID for an Experience Cloud site, internal, or `null`.

**groupId**
Type: *String*
The ID of the group.

**fileUpload**
Type: *ConnectApi.BinaryInput*
File to use as the photo. The content type must be usable as an image.
Return Value
Type: `ConnectApi.BannerPhoto`

Usage
This method is successful only when the context user is the group manager or owner, or has Modify All Data permission. Photos are processed asynchronously and might not be visible right away.

```java
setBannerPhotoWithAttributes(communityId, groupId, bannerPhoto)
```
Set and crop an uploaded file as the group banner photo.

API Version
36.0

Requires Chatter
Yes

Signature
```java
public static ConnectApi.BannerPhoto setBannerPhotoWithAttributes(String communityId, String groupId, ConnectApi.BannerPhotoInput bannerPhoto)
```

Parameters
- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.
- `groupId`
  Type: `String`
  The ID of the group.
- `bannerPhoto`
  Type: `ConnectApi.BannerPhotoInput`
  A `ConnectApi.BannerPhotoInput` object that specifies the ID and version of the file, and how to crop the file.

Return Value
Type: `ConnectApi.BannerPhoto`

Usage
This method is successful only when the context user is the group manager or owner, or has Modify All Data permission. Photos are processed asynchronously and might not be visible right away.
setBannerPhotoWithAttributes(communityId, groupId, bannerPhoto, fileUpload)

Set and crop a file that hasn’t been uploaded as the group banner photo.

API Version
36.0

Requires Chatter
Yes

Signature
public static ConnectApi.BannerPhoto setBannerPhotoWithAttributes(String communityId, String groupId, ConnectApi.BannerPhotoInput bannerPhoto, ConnectApi.BinaryInput fileUpload)

Parameters
communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

groupId
  Type: String
  The ID of the group.

bannerPhoto
  Type: ConnectApi.BannerPhotoInput
  A ConnectApi.BannerPhotoInput object specifying the cropping parameters.

fileUpload
  Type: ConnectApi.BinaryInput
  File to use as the photo. The content type must be usable as an image.

Return Value
Type: ConnectApi.BannerPhoto

Usage
This method is successful only when the context user is the group manager or owner, or has Modify All Data permission.
Photos are processed asynchronously and might not be visible right away.

setPhoto(communityId, groupId, fileId, versionNumber)

Set an uploaded file as the group photo.
API Version

28.0

Requires Chatter

Yes

Signature

public static ConnectApi.Photo setPhoto(String communityId, String groupId, String fileId, Integer versionNumber)

Parameters

communityId
    Type: String
    ID for an Experience Cloud site, internal, or null.

groupId
    Type: String
    ID for a group.

fileId
    Type: String
    ID of a file already uploaded. The key prefix must be 069, and the file must be an image that is smaller than 2 GB.

versionNumber
    Type: Integer
    Version number of the existing file. Specify either an existing version number or, to get the latest version, specify null.

Return Value

Type: ConnectApi.Photo

Usage

This method is successful only when the context user is the group manager or owner, or has Modify All Data permission.

Photos are processed asynchronously and might not be visible right away.

Sample: Updating a Group Photo with an Existing File

When a group is created, it doesn’t have a group photo. You can set an existing photo that has already been uploaded to Salesforce as the group photo. The key prefix must be 069 and the file size must be less than 2 GB.

    String communityId = null;
    ID groupId = '0F9x00000000hAK';
    ID fileId = '069x00000001Ion';

    // Set photo
    ConnectApi.Photo photo = ConnectApi.ChatterGroups.setPhoto(communityId, groupId, fileId, null);
**setPhoto(communityId, groupId, fileUpload)**

Set a file that hasn’t been uploaded as the group photo.

**API Version**

28.0

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.Photo setPhoto(String communityId, String groupId, ConnectApi.BinaryInput fileUpload)
```

**Parameters**

- **communityId**
  Type: String
  ID for an Experience Cloud site, internal, or null.

- **groupId**
  Type: String
  ID for a group.

- **fileUpload**
  Type: ConnectApi.BinaryInput
  File to use as the photo. The content type must be usable as an image.

**Return Value**

Type: ConnectApi.Photo

**Usage**

This method is successful only when the context user is the group manager or owner, or has Modify All Data permission. Photos are processed asynchronously and might not be visible right away.

**Sample: Uploading a New File and Using It as a Group Photo**

When a group is created, it doesn’t have a group photo. You can upload a photo and set it as the group photo.

```java
String communityId = null;
ID groupId = '0F9x00000000hAP';
ID photoId = '069x00000001Io0';

// Set photo
List<ContentVersion> groupPhoto = [Select c.VersionData From ContentVersion c where ContentDocumentId=:photoId];
ConnectApi.BinaryInput binary = new ConnectApi.BinaryInput(groupPhoto.get(0).VersionData,
```
ConnectApi.Photo photo = ConnectApi.ChatterGroups.setPhoto(communityId, groupId, binary);

setPhotoWithAttributes(communityId, groupId, photo)
Set and crop an uploaded file as the group photo.

API Version
29.0

Requires Chatter
Yes

Signature
public static ConnectApi.Photo setPhotoWithAttributes(String communityId, String groupId, ConnectApi.PhotoInput photo)

Parameters
communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

groupId
  Type: String
  ID for a group.

photo
  Type: ConnectApi.PhotoInput
  A ConnectApi.PhotoInput object that specifies the ID and version of the file, and how to crop the file.

Return Value
Type: ConnectApi.Photo

Usage
This method is successful only when the context user is the group manager or owner, or has Modify All Data permission.
Photos are processed asynchronously and might not be visible right away.

setPhotoWithAttributes(communityId, groupId, photo, fileUpload)
Set and crop a file that hasn’t been uploaded as the group photo.
API Version
29.0

Requires Chatter
Yes

Signature

public static ConnectApi.PhotosetPhotoWithAttributes(String communityId, String groupId, ConnectApi.PhotoInput photo, ConnectApi.BinaryInput fileUpload)

Parameters

communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

groupId
  Type: String
  ID for a group.

photo
  Type: ConnectApi.PhotoInput
  A ConnectApi.PhotoInput object that specifies how to crop the file specified in fileUpload.

fileUpload
  Type: ConnectApi.BinaryInput
  File to use as the photo. The content type must be usable as an image.

Return Value

Type: ConnectApi.Photo

Usage

This method is successful only when the context user is the group manager or owner, or has Modify All Data permission.
Photos are processed asynchronously and might not be visible right away.

updateGroup(communityId, groupId, groupInput)

Update the settings of a group.

API Version
28.0

Requires Chatter
Yes
Signature

public static ConnectApi.ChatterGroup updateGroup(String communityId, String groupId, ConnectApi.ChatterGroupInput groupInput)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

groupId
Type: String
ID for a group.

groupInput
Type: ConnectApi.ChatterGroupInput
A ConnectApi.ChatterGroupInput object.

Return Value
Type: ConnectApi.ChatterGroup

Usage
This method is successful only when the context user is the group manager or owner, or has Modify All Data permission. Use this method to update any settings in the ConnectApi.ChatterGroupInput class. These settings include the group title and text in the “Information” section, whether the group is public or private, and whether the group is archived.

Example
This example archives a group.

```java
String groupId = '0F9D00000000qSz';
String communityId = null;
ConnectApi.ChatterGroupInput groupInput = new ConnectApi.ChatterGroupInput();
groupInput.isArchived = true;
ConnectApi.ChatterGroups.updateGroup(communityId, groupId, groupInput);
```

updateGroupMember(communityId, membershipId, role)
Update the role of a group member.

API Version
29.0

Requires Chatter
Yes
Signature

```java
public static ConnectApi.ChatterGroup updateGroupMember(String communityId, String membershipId, ConnectApi.GroupMembershipType role)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- `membershipId`
  - Type: `String`
  - ID for a membership.

- `role`
  - Type: `ConnectApi.GroupMembershipType`
  - The group membership type. One of these values:
    - `GroupManager`
    - `StandardMember`

Return Value

Type: `ConnectApi.ChatterGroup`

Usage

This method is successful only when the context user is the group manager or owner, or has Modify All Data permission.

```java
updateMyChatterSettings(communityId, groupId, emailFrequency)
```

Update the context user's email frequency for a group.

API Version

28.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.GroupChatterSettings updateMyChatterSettings(String communityId, String groupId, ConnectApi.GroupEmailFrequency emailFrequency)
```

Parameters

- `communityId`
  - Type: `String`
ID for an Experience Cloud site, internal, or null.

**groupId**
Type: String
ID for a group.

**emailFrequency**
Type: `ConnectApi.GroupEmailFrequency`
Frequency with which a user receives email.
- EachPost
- DailyDigest
- WeeklyDigest
- Never
- UseDefault
The value UseDefault uses the value set in a call to `updateChatterSettings(communityId, userId, defaultGroupEmailFrequency)`.

**Return Value**
Type: `ConnectApi.GroupChatterSettings`

**updateRequestStatus(communityId, requestId, status)**
Update a request to join a private group.

**API Version**
28.0

**Requires Chatter**
Yes

**Signature**

```java
public static ConnectApi.GroupMembershipRequest updateRequestStatus(String communityId, String requestId, ConnectApi.GroupMembershipRequestStatus status)
```

**Parameters**

**communityId**
Type: String
ID for an Experience Cloud site, internal, or null.

**requestId**
Type: String
ID for a request to join a private group.

**status**
Type: `ConnectApi.GroupMembershipRequestStatus`
Status of the request:
• Accepted
• Declined

The Pending value of the enum is not valid in this method.

Return Value
Type: ConnectApi.GroupMembershipRequest

Usage
This method is successful only when the context user is the group manager or owner, or has Modify All Data permission.

Sample: Accepting or Declining a Request to Join a Private Group
This sample code calls ConnectApi.ChatterGroups.updateRequestStatus and passes it the membership request ID and an ConnectApi.GroupMembershipRequestStatus.Accepted status. You can also pass ConnectApi.GroupMembershipRequestStatus.Declined.

```java
String communityId = null;
ID groupId = '0F9x00000000hAZ';
String requestId = '0I5x000000001snCAA';

ConnectApi.GroupMembershipRequest membershipRequestRep =
ConnectApi.ChatterGroups.updateRequestStatus(communityId, requestId,
ConnectApi.GroupMembershipRequestStatus.Accepted);
```

updateRequestStatus(communityId, requestId, status, responseMessage)
Update a request to join a private group and optionally provide a message when the request is denied.

API Version
35.0

Requires Chatter
Yes

Signature
public static ConnectApi.GroupMembershipRequest updateRequestStatus(String communityId,
String requestId, ConnectApi.GroupMembershipRequestStatus status, String responseMessage)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.
requestId
Type: String
ID for a request to join a private group.

status
Type: ConnectApi.GroupMembershipRequestStatus
Status of the request:
• Accepted
• Declined
The Pending value of the enum is not valid in this method.

responseMessage
Type: String
Provide a message to the user if their membership request is declined. The value of this property is used only when the value of the status property is Declined.
The maximum length is 756 characters.

Return Value
Type: ConnectApi.GroupMembershipRequest

Usage
This method is successful only when the context user is the group manager or owner, or has Modify All Data permission.

ChatterGroups Test Methods
The following are the test methods for ChatterGroups. All methods are static.
For information about using these methods to test your ConnectApi code, see Testing ConnectApi Code.

setTestSearchGroups(communityId, q, result)
Register a ConnectApi.ChatterGroupPage object to be returned when the matching ConnectApi.searchGroups method is called in a test context. Use the test method with the same parameters or you receive an exception.

API Version
29.0

Signature
public static Void setTestSearchGroups(String communityId, String q, ConnectApi.ChatterGroupPage result)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards. Can be specified as null.

result
Type: ConnectApi.ChatterGroupPage
Test ConnectApi.ChatterGroupPage object.

Return Value
Type: Void

SEE ALSO:
searchGroups(communityId, q)

setTestSearchGroups(communityId, q, pageParam, pageSize, result)

Register a ConnectApi.ChatterGroupPage object to be returned when the matching ConnectApi.searchGroups method is called in a test context. Use the test method with the same parameters or you receive an exception.

API Version
28.0

Signature
public static Void setTestSearchGroups(String communityId, String q, Integer pageParam, Integer pageSize, ConnectApi.ChatterGroupPage result)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards. Can be specified as null.

pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

```java
result
    Type: ConnectApi.ChatterGroupPage
    Test ConnectApi.ChatterGroupPage object.
```

Return Value
Type: Void

SEE ALSO:
- `searchGroups(communityId, q, pageParam, pageSize)`

```java
setTestSearchGroups(communityId, q, archiveStatus, pageParam, pageSize, result)
```

Register a ConnectApi.ChatterGroupPage object to be returned when the matching ConnectApi.searchGroups method is called in a test context. Use the test method with the same parameters or you receive an exception.

API Version
29.0

Signature
```java
public static Void setTestSearchGroups(String communityId, String q,
ConnectApi.GroupArchiveStatus, archiveStatus, Integer pageParam, Integer pageSize,
ConnectApi.ChatterGroupPage result)
```

Parameters
- `communityId`
  Type: String
  ID for an Experience Cloud site, internal, or null.
- `q`
  Type: String
  Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards. Can be specified as null.
- `archiveStatus`
  Type: ConnectApi.GroupArchiveStatus
  Archive status of groups.
  - All—all groups, including groups that are archived and groups that aren’t archived.
  - Archived—Groups that are archived.
  - NotArchived—Groups that aren’t archived.
pageParam  
Type: Integer  
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.  

pageSize  
Type: Integer  
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.  

result  
Type: ConnectApi.ChatterGroupPage  
Test ConnectApi.ChatterGroupPage object.  

Return Value  
Type: Void  

SEE ALSO:  
searchGroups(communityId, q, archiveStatus, pageParam, pageSize)  

ChatterMessages Class  
Get, send, search, and reply to private messages. You can also get and search private conversations, mark conversations as read, and get a count of unread private messages.  

Namespace  
ConnectApi  

Usage  
Private messages and direct messages are different features. Direct messages are newer and offer a richer feature set for private communication in Experience Cloud sites. Direct messages are a capability within Chatter feeds. To work with direct messages, use the ChatterFeeds Class.  

ChatterMessages Methods  
The following are methods for ChatterMessages. All methods are static.  

IN THIS SECTION:  
getConversation(conversationId)  
Get a conversation.  
getConversation(conversationId, pageParam, pageSize)  
Get a page of a conversation.  
getConversation(communityId, conversationId)  
Get a conversation from an Experience Cloud site.
getConversation(communityId, conversationId, pageParam, pageSize)
Get a page of a conversation from an Experience Cloud site.

getConversations()
Get the most recent conversations.

getConversations(pageParam, pageSize)
Get a page of conversations.

getConversations(communityId)
Get the most recent conversations from an Experience Cloud site.

getConversations(communityId, pageParam, pageSize)
Get a page of conversations from an Experience Cloud site.

ggetMessage(messageId)
Get a message.

ggetMessage(communityId, messageId)
Get a message from an Experience Cloud site.

getMessages()
Get the most recent messages.

getMessages(pageParam, pageSize)
Get a page of messages.

getMessages(communityId)
Get the most recent messages from an Experience Cloud site.

getMessages(communityId, pageParam, pageSize)
Get a page of messages from an Experience Cloud site.

getUnreadCount()
Get the number of conversations that are marked unread.

getUnreadCount(communityId)
Get the number of conversations that are marked unread in an Experience Cloud site.

markConversationRead(conversationId, read)
Mark a conversation as read or unread.

markConversationRead(communityId, conversationID, read)
Mark a conversation as read or unread in an Experience Cloud site.

replyToMessage(text, inReplyTo)
Reply to a message.

replyToMessage(communityId, text, inReplyTo)
Reply to a message in an Experience Cloud site.

searchConversation(conversationId, q)
Get a conversation that matches the search criteria.

searchConversation(conversationId, pageParam, pageSize, q)
Get a conversation with a page of messages that match the search criteria.

searchConversation(communityId, conversationId, q)
Get a conversation with messages that match the search criteria in an Experience Cloud site.
searchConversation(communityId, conversationId, pageParam, pageSize, q)
Get a conversation with a page of messages that match the search criteria in an Experience Cloud site.

searchConversations(q)
Get conversations in which member names and messages match the search criteria.

searchConversations(pageParam, pageSize, q)
Get a page of conversations in which member names and messages match the search criteria.

searchConversations(communityId, q)
Get conversations in which member names and messages match the search criteria in an Experience Cloud site.

searchConversations(communityId, pageParam, pageSize, q)
Get a page of conversations in which member names and messages match the search criteria in an Experience Cloud site.

searchMessages(q)
Get messages that match the search criteria.

searchMessages(pageParam, pageSize, q)
Get a page of messages that match the search criteria.

searchMessages(communityId, q)
Get messages that match the search criteria in an Experience Cloud site.

searchMessages(communityId, pageParam, pageSize, q)
Get a page of messages that match the search criteria in an Experience Cloud site.

sendMessage(text, recipients)
Send a message to a list of recipients.

sendMessage(communityId, text, recipients)
Send a message to a list of recipients in an Experience Cloud site.

getConversation(conversationId)
Get a conversation.

API Version
29.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterConversation getConversation(String conversationId)

Parameters
conversationId
Type: String
ID for the conversation.
Return Value
Type: ConnectApi.ChatterConversation

getConversation(conversationId, pageParam, pageSize)
Get a page of a conversation.

API Version
29.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterConversation getConversation(String conversationId, String pageParam, Integer pageSize)

Parameters
conversationId
Type: String
ID for the conversation.

pageParam
Type: String
Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

Return Value
Type: ConnectApi.ChatterConversation

getConversation(communityId, conversationId)
Get a conversation from an Experience Cloud site.

API Version
30.0

Requires Chatter
Yes
Signature

```java
public static ConnectApi.ChatterConversation getConversation(String communityId, String conversationId)
```

Parameters

- `communityId`: Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.
- `conversationId`: Type: `String`
  - ID for the conversation.

Return Value

Type: `ConnectApi.ChatterConversation`

```java
getConversation(communityId, conversationId, pageParam, pageSize)
```

Get a page of a conversation from an Experience Cloud site.

API Version

30.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.ChatterConversation getConversation(String communityId, String conversationId, String pageParam, String pageSize)
```

Parameters

- `communityId`: Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.
- `conversationId`: Type: `String`
  - ID for the conversation.
- `pageParam`: Type: `String`
  - Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.
**pageSize**
Type: Integer

Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

**Return Value**
Type: ConnectApi.ChatterConversation

**getConversations()**
Get the most recent conversations.

**API Version**
29.0

**Requires Chatter**
Yes

**Signature**
public static ConnectApi.ChatterConversationPage getConversations()

**Return Value**
Type: ConnectApi.ChatterConversationPage

**getConversations(pageParam, pageSize)**
Get a page of conversations.

**API Version**
29.0

**Requires Chatter**
Yes

**Signature**
public static ConnectApi.ChatterConversationPage getConversations(String pageParam, Integer pageSize)

**Parameters**

*pageParam*  
Type: String
Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

Return Value
Type: ConnectApi.ChatterConversationPage

getConversations(communityId)
Get the most recent conversations from an Experience Cloud site.

API Version
30.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterConversationPage getConversations(String communityId)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

Return Value
Type: ConnectApi.ChatterConversationPage

getConversations(communityId, pageParam, pageSize)
Get a page of conversations from an Experience Cloud site.

API Version
30.0

Requires Chatter
Yes
public static ConnectApi.ChatterConversationPage getConversations(String communityId, String pageParam, Integer pageSize)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

pageParam
Type: String
Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

Return Value
Type: ConnectApi.ChatterConversationPage

getMessage (messageId)
Get a message.

API Version
29.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterMessage getMessage(String messageId)

Parameters

messageId
Type: String
ID for the message.

Return Value
Type: ConnectApi.ChatterMessage
getMessage(communityId, messageId)
Get a message from an Experience Cloud site.

API Version
30.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterMessage getMessage(String communityId, String messageId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

messageId
Type: String
ID for the message.

Return Value
Type: ConnectApi.ChatterMessage

getMessages()
Get the most recent messages.

API Version
29.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterMessagePage getMessages()

Return Value
Type: ConnectApi.ChatterMessagePage
**getMessages(pageParam, pageSize)**

Get a page of messages.

**API Version**

29.0

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.ChatterMessagePage getMessages(String pageParam, Integer pageSize)
```

**Parameters**

- `pageParam`
  - Type: `String`
  - Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

- `pageSize`
  - Type: `Integer`
  - Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**Return Value**

Type: `ConnectApi.ChatterMessagePage`

**getMessages(communityId)**

Get the most recent messages from an Experience Cloud site.

**API Version**

30.0

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.ChatterMessagePage getMessages(String communityId)
```
Parameters

*communityId*
Type: String
ID for an Experience Cloud site, internal, or null.

Return Value
Type: `ConnectApi.ChatterMessagePage`

**getMessages(communityId, pageParam, pageSize)**
Get a page of messages from an Experience Cloud site.

API Version
30.0

Requires Chatter
Yes

Signature

```java
public static ConnectApi.ChatterMessagePage getMessages(String communityId, String pageParam, Integer pageSize)
```

Parameters

*communityId*
Type: String
ID for an Experience Cloud site, internal, or null.

*pageParam*
Type: String
Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in null, the first page is returned.

*pageSize*
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

Return Value
Type: `ConnectApi.ChatterMessagePage`

**getUnreadCount()**
Get the number of conversations that are marked unread.
API Version
29.0

Requires Chatter
Yes

Signature
public static ConnectApi.UnreadConversationCount getUnreadCount()

Return Value
Type: ConnectApi.UnreadConversationCount
If there are fewer than 50 unread conversations, ConnectApi.UnreadConversationCount returns the exact number of unread conversations and the hasMore property is false. If there are more than 50 unread conversations, ConnectApi.UnreadConversationCount returns 50 unread conversations and the hasMore property is true.

Example
ConnectApi.UnreadConversationCount unread = ConnectApi.ChatterMessages.getUnreadCount();

getUnreadCount(communityId)
Get the number of conversations that are marked unread in an Experience Cloud site.

API Version
30.0

Requires Chatter
Yes

Signature
public static ConnectApi.UnreadConversationCount getUnreadCount(String communityId)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

Return Value
Type: ConnectApi.UnreadConversationCount
If there are fewer than 50 unread conversations, `ConnectApi.UreadConversationCount` returns the exact number of unread conversations and the `hasMore` property is `false`. If there are more than 50 unread conversations, `ConnectApi.UreadConversationCount` returns 50 unread conversations and the `hasMore` property is `true`.

### markConversationRead(conversationId, read)
Mark a conversation as read or unread.

#### API Version
29.0

#### Requires Chatter
Yes

#### Signature
```java
public static ConnectApi.ChatterConversationSummary markConversationRead(String conversationId, Boolean read)
```

#### Parameters
- **conversationId**
  - Type: `String`
  - ID for the conversation.
- **read**
  - Type: `Boolean`
  - Specify whether the conversation is read (`true`) or not (`false`).

#### Return Value
Type: `ConnectApi.ChatterConversationSummary`

### markConversationRead(communityId, conversationID, read)
Mark a conversation as read or unread in an Experience Cloud site.

#### API Version
30.0

#### Requires Chatter
Yes

#### Signature
```java
public static ConnectApi.ChatterConversationSummary markConversationRead(String communityId, String conversationID, Boolean read)
```
Parameters

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **conversationId**
  - Type: `String`
  - ID for the conversation.

- **read**
  - Type: `Boolean`
  - Specify whether the conversation is read (`true`) or not (`false`).

Return Value

Type: `ConnectApi.ChatterConversationSummary`

**replyToMessage(text, inReplyTo)**

Reply to a message.

API Version

29.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.ChatterMessage replyToMessage(String text, String inReplyTo)
```

Parameters

- **text**
  - Type: `String`
  - Text of the message. Can't be empty or over 10,000 characters.

- **inReplyTo**
  - Type: `String`
  - ID of the message that is being responded to.

Return Value

Type: `ConnectApi.ChatterMessage`

**replyToMessage(communityId, text, inReplyTo)**

Reply to a message in an Experience Cloud site.
API Version
30.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterMessage replyToMessage(String communityId, String text, String inReplyTo)

Parameters
communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.
text
  Type: String
  Text of the message. Can’t be empty or over 10,000 characters.
inReplyTo
  Type: String
  ID of the message that is being responded to.

Return Value
Type: ConnectApi.ChatterMessage

searchConversation(conversationId, q)
Get a conversation that matches the search criteria.

API Version
29.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterConversation searchConversation(String conversationId, String q)
Parameters

conversationId
  Type: String
  ID for the conversation.

q
  Type: String
  Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

Return Value

Type: ConnectApi.ChatterConversation

searchConversation(conversationId, pageParam, pageSize, q)

Get a conversation with a page of messages that match the search criteria.

API Version

29.0

Requires Chatter

Yes

Signature

public static ConnectApi.ChatterConversation searchConversation(String conversationId, String pageParam, Integer pageSize, String q)

Parameters

conversationId
  Type: String
  ID for the conversation.

pageParam
  Type: String
  Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
  Type: Integer
  Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

q
  Type: String
  Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.
Return Value
Type: `ConnectApi.ChatterConversation`

`searchConversation(communityId, conversationId, q)`
Get a conversation with messages that match the search criteria in an Experience Cloud site.

API Version
30.0

Requires Chatter
Yes

Signature
`public static ConnectApi.ChatterConversation searchConversation(String communityId, String conversationId, String q)`

Parameters
`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

`conversationId`
Type: `String`
ID for the conversation.

`q`
Type: `String`
Required and can't be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See `Wildcards`.

Return Value
Type: `ConnectApi.ChatterConversation`

`searchConversation(communityId, conversationId, pageParam, pageSize, q)`
Get a conversation with a page of messages that match the search criteria in an Experience Cloud site.

API Version
30.0

Requires Chatter
Yes
Signature

```java
public static ConnectApi.ChatterConversation searchConversation(String communityId,
                    String conversationId, String pageParam, Integer pageSize, String q)
```

Parameters

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **conversationId**
  - Type: `String`
  - ID for the conversation.

- **pageParam**
  - Type: `String`
  - Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

- **pageSize**
  - Type: `Integer`
  - Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

- **q**
  - Type: `String`
  - Required and can’t be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See [Wildcards](#).

Return Value

- Type: `ConnectApi.ChatterConversation`

`searchConversations(q)`

Get conversations in which member names and messages match the search criteria.

API Version

29.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.ChatterConversationPage searchConversations(String q)
```
Parameters

$q$
Type: String

Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

Return Value

Type: ConnectApi.ChatterConversationPage

searchConversations(pageParam, pageSize, q)

Get a page of conversations in which member names and messages match the search criteria.

API Version

29.0

Requires Chatter

Yes

Signature

public static ConnectApi.ChatterConversationPage searchConversations(String pageParam, Integer pageSize, String q)

Parameters

$\text{pageParam}$
Type: String

Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in null, the first page is returned.

$\text{pageSize}$
Type: Integer

Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

$q$
Type: String

Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

Return Value

Type: ConnectApi.ChatterConversationPage

searchConversations(communityId, q)

Get conversations in which member names and messages match the search criteria in an Experience Cloud site.
API Version
30.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterConversationPage searchConversations(String communityId, String q)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

Return Value
Type: ConnectApi.ChatterConversationPage

searchConversations(communityId, pageParam, pageSize, q)
Get a page of conversations in which member names and messages match the search criteria in an Experience Cloud site.

API Version
30.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterConversationPage searchConversations(String communityId, String pageParam, Integer pageSize, String q)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.
**searchMessages**(q)

Get messages that match the search criteria.

### API Version

29.0

### Requires Chatter

Yes

### Signature

```java
public static ConnectApi.ChatterMessagePage searchMessages(String q)
```

### Parameters

- **q**
  - Type: `String`
  - Required and can't be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See [Wildcards](#).

### Return Value

Type: `ConnectApi.ChatterMessagePage`
API Version
29.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterMessagePage searchMessages(String pageParam, Integer pageSize, String q)

Parameters

pageParam
Type: String
Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

q
Type: String
Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

Return Value
Type: ConnectApi.ChatterMessagePage

searchMessages(communityId, q)
Get messages that match the search criteria in an Experience Cloud site.

API Version
30.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterMessagePage searchMessages(String communityId, String q)
Parameters

**communityId**
Type: **String**
ID for an Experience Cloud site, internal, or **null**.

**q**
Type: **String**
Required and can't be **null**. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See [Wildcards](#).

Return Value
Type: **ConnectApi.ChatterMessagePage**

`searchMessages(communityId, pageParam, pageSize, q)`
Get a page of messages that match the search criteria in an Experience Cloud site.

API Version
30.0

Requires Chatter
Yes

Signature

```
public static ConnectApi.ChatterMessagePage searchMessages(String communityId, String pageParam, Integer pageSize, String q)
```

Parameters

**communityId**
Type: **String**
ID for an Experience Cloud site, internal, or **null**.

**pageParam**
Type: **String**
Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as `currentPageToken` or `nextPageToken`. If you pass in **null**, the first page is returned.

**pageSize**
Type: **Integer**
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in **null**, the default size is 25.

**q**
Type: **String**
Required and can't be **null**. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See [Wildcards](#).
Return Value
Type: `ConnectApi.ChatterMessagePage`

`sendMessage(text, recipients)`
Send a message to a list of recipients.

API Version
29.0

Requires Chatter
Yes

Signature
```java
public static ConnectApi.ChatterMessage sendMessage(String text, String recipients)
```

Parameters

text
Type: `String`
Text of the message. Can't be empty or over 10,000 characters.

recipients
Type: `String`
Up to nine comma-separated IDs of the users receiving the message.

Return Value
Type: `ConnectApi.ChatterMessage`

`sendMessage(communityId, text, recipients)`
Send a message to a list of recipients in an Experience Cloud site.

API Version
30.0

Requires Chatter
Yes

Signature
```java
public static ConnectApi.ChatterMessage sendMessage(String communityId, String text, String recipients)
```
Parameters

- **communityId**: String
  - ID for an Experience Cloud site, internal, or null.

- **text**: String
  - Text of the message. Can’t be empty or over 10,000 characters.

- **recipients**: String
  - Up to nine comma-separated IDs of users to receive the message.

Return Value

Type: `ConnectApi.ChatterMessage`

ChatterUsers Class

Access information about users, such as activity, followers, subscriptions, files, and groups.

Namespace

- `ConnectApi`

ChatterUsers Methods

The following are methods for ChatterUsers. All methods are static.

IN THIS SECTION:

- `exportUserActivities(communityId, userId)`: Export Chatter-related user activity, such as bookmarks, topic endorsements, and votes.
- `follow(communityId, userId, subjectId)`: Follow a user or record.
- `getChatterSettings(communityId, userId)`: Get the default Chatter settings for a user.
- `getFollowers(communityId, userId)`: Get the first page of followers for a user.
- `getFollowers(communityId, userId, pageParam, pageSize)`: Get a page of followers for a user.
- `getFollowings(communityId, userId)`: Get the first page of users and records that a user follows.
- `getFollowings(communityId, userId, pageParam)`: Get a page of users and records that a user follows.
getFollowings(communityId, userId, pageParam, pageSize)
Get a page with the specified number of users and records that a user follows.

getFollowings(communityId, userId, filterType)
Get the first page of records, filtered by key prefix, that a user follows.

getFollowings(communityId, userId, filterType, pageParam)
Get a page of records, filtered by key prefix, that a user follows.

getFollowings(communityId, userId, filterType, pageParam, pageSize)
Get a page with the specified number of records, filtered by key prefix, that a user follows.

getReputation(communityId, userId)
Get a user’s reputation.

getUser(communityId, userId)
Get information about a user.

getUserBatch(communityId, userIds)
Get information about a list of users.

getUserGroups(communityId, userId)
Get a user’s groups.

getUserGroups(communityId, userId, pageParam, pageSize)
Get a page of a user’s groups.

getUsers(communityId)
Get the first page of users.

getUsers(communityId, pageParam, pageSize)
Get a page of users.

purgeUserActivities(communityId, userId)
Start a job to purge Chatter-related user activity, such as bookmarks, topic endorsements, and votes.

searchUserGroupDetails(communityId, userId, q)
Get the user’s groups that match the search criteria.

searchUserGroupDetails(communityId, userId, q, pageParam, pageSize)
Get a page of a user’s groups that match the search criteria.

searchUsers(communityId, q)
Get the first page of users that match the search criteria.

searchUsers(communityId, q, pageParam, pageSize)
Get a page of users that match the search criteria.

searchUsers(communityId, q, searchContextId, pageParam, pageSize)
Get a page of users that match the search criteria.

updateChatterSettings(communityId, userId, defaultGroupEmailFrequency)
Update the default Chatter settings for a user.

updateUser(communityId, userId, userInput)
Update the About Me section for a user.
exportUserActivities(communityId, userId)

Export Chatter-related user activity, such as bookmarks, topic endorsements, and votes.

API Version

42.0

Requires Chatter

Yes

Signature

public static ConnectApi.UserActivitiesJob exportUserActivities(String communityId, String userId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

userId
Type: String
ID of the user.

Return Value

Type: ConnectApi.UserActivitiesJob

Usage

The following activities can be exported.

• Bookmark—User bookmarked a post.
• ChatterActivity—Total counts of posts and comments made and likes and comments received for a user.
• ChatterLike—User liked a post or comment.
• Comment—User commented on a post.
• CompanyVerify—User verified comment.
• DownVote—User downvoted a post or comment.
• FeedEntityRead—User read a post.
• FeedRead—User read a feed.
• Mute—User muted a post.
• Post—User made a post.
• TopicEndorsement—User endorsed another user on a topic or received endorsement on a topic.
• UpVote—User upvoted a post or comment.
follow(communityId, userId, subjectId)
Follow a user or record.

API Version
28.0

Requires Chatter
Yes

Signature
public static ConnectApi.Subscription follow(String communityId, String userId, String subjectId)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

userId
Type: String
ID for the context user or the keyword me.

subjectId
Type: String
ID of the user or record to follow.

Return Value
Type: ConnectApi.Subscription

Example
ChatterUsers.ConnectApi.Subscription subscriptionToRecord =
ConnectApi.ChatterUsers.follow(null, 'me', '001RR000002G4Y0');

SEE ALSO:
   Unfollow a Record

getChatterSettings(communityId, userId)
Get the default Chatter settings for a user.

API Version
28.0
Requires Chatter
Yes

Signature
public static ConnectApi.UserChatterSettings getChatterSettings(String communityId, String userId)

Parameters
communityId
   Type: String
   ID for an Experience Cloud site, internal, or null.

userId
   Type: String
   ID for the context user or the keyword me.

Return Value
Type: ConnectApi.UserChatterSettings

getFollowers(communityId, userId)
Get the first page of followers for a user.

API Version
28.0

Available to Guest Users
32.0

Requires Chatter
Yes

Signature
public static ConnectApi.FollowerPage getFollowers(String communityId, String userId)

Parameters
communityId
   Type: String
   ID for an Experience Cloud site, internal, or null.

userId
   Type: String
ID for a user.

Return Value
Type: ConnectApi.FollowerPage

getFollowers(communityId, userId, pageParam, pageSize)
Get a page of followers for a user.

API Version
28.0

Available to Guest Users
32.0

Requires Chatter
Yes

Signature
public static ConnectApi.FollowerPage getFollowers(String communityId, String userId, Integer pageParam, Integer pageSize)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

userId
Type: String
ID for a user.

pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

Return Value
Type: ConnectApi.FollowerPage
**getFollowings(communityId, userId)**
Get the first page of users and records that a user follows.

**API Version**
28.0

**Available to Guest Users**
32.0

**Requires Chatter**
Yes

**Signature**
```java
public static ConnectApi.FollowingPage getFollowings(String communityId, String userId)
```

**Parameters**
- **communityId**
  Type: String
  ID for an Experience Cloud site, internal, or null.
- **userId**
  Type: String
  ID for a user.

**Return Value**
Type: `ConnectApi.FollowingPage`

**getFollowings(communityId, userId, pageParam)**
Get a page of users and records that a user follows.

**API Version**
28.0

**Available to Guest Users**
32.0

**Requires Chatter**
Yes
Signature

```java
public static ConnectApi.FollowingPage getFollowings(String communityId, String userId, Integer pageParam)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- `userId`
  - Type: `String`
  - ID for a user.

- `pageParam`
  - Type: `Integer`
  - Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

Return Value

Type: `ConnectApi.FollowingPage`

```java
getFollowings(communityId, userId, pageParam, pageSize)
```

Get a page with the specified number of users and records that a user follows.

API Version

28.0

Available to Guest Users

32.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.FollowingPage getFollowings(String communityId, String userId, Integer pageParam, Integer pageSize)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- `userId`
  - Type: `String`

- `pageParam`
  - Type: `Integer`

- `pageSize`
  - Type: `Integer`
ID for a user.

**pageParam**

Type: **Integer**

Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

**pageSize**

Type: **Integer**

Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**Return Value**

Type: `ConnectApi.FollowingPage`

**getFollowings(communityId, userId, filterType)**

Get the first page of records, filtered by key prefix, that a user follows.

**API Version**

28.0

**Available to Guest Users**

32.0

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.FollowingPage getFollowings(String communityId, String userId, String filterType)
```

**Parameters**

**communityId**

Type: **String**

ID for an Experience Cloud site, internal, or `null`.

**userId**

Type: **String**

ID for a user.

**filterType**

Type: **String**

Specifies the key prefix to filter the type of objects returned. A key prefix is the first three characters of the object ID, which specifies the object type. For example, User objects have a prefix of `00S` and Group objects have a prefix of `0F9`. 
Return Value
Type: ConnectApi.FollowingPage

getFollowings(communityId, userId, filterType, pageParam)
Get a page of records, filtered by key prefix, that a user follows.

API Version
28.0

Available to Guest Users
32.0

Requires Chatter
Yes

Signature
public static ConnectApi.FollowingPage getFollowings(String communityId, String userId, String filterType, Integer pageParam)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

userId
Type: String
ID for a user.

filterType
Type: String
Specifies the key prefix to filter the type of objects returned. A key prefix is the first three characters of the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.

pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

Return Value
Type: ConnectApi.FollowingPage

getFollowings(communityId, userId, filterType, pageParam, pageSize)
Get a page with the specified number of records, filtered by key prefix, that a user follows.
API Version
28.0

Available to Guest Users
32.0

Requires Chatter
Yes

Signature
```
public static ConnectApi.FollowingPage getFollowings(String communityId, String userId,
String filterType, Integer pageParam, Integer pageSize)
```

Parameters
- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.
- **userId**
  - Type: String
  - ID for a user.
- **filterType**
  - Type: String
  - Specifies the key prefix to filter the type of objects returned. A key prefix is the first three characters of the object ID, which specifies the object type. For example, User objects have a prefix of 005 and Group objects have a prefix of 0F9.
- **pageParam**
  - Type: Integer
  - Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.
- **pageSize**
  - Type: Integer
  - Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

Return Value
Type: ConnectApi.FollowingPage

**getReputation(communityId, userId)**
Get a user's reputation.
Available to Guest Users
32.0

Requires Chatter
Yes

Signature
public static ConnectApi.Reputation getReputation(String communityId, String userId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

userId
Type: String
ID of the user.

Return Value
Type: ConnectApi.Reputation

getUser(communityId, userId)
Get information about a user.

API Version
28.0

Available to Guest Users
32.0

Requires Chatter
Yes

Signature
public static ConnectApi.UserSummary getUser(String communityId, String userId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.
userId
    Type: String
    ID for a user.

Return Value
Type: ConnectApi.UserDetail

Usage
If the user is external, the properties that the ConnectApi.UserDetail output class shares with the ConnectApi.UserSummary output class can have non-null values. Other properties are always null.

getUserBatch(communityId, userIds)
Get information about a list of users.

API Version
31.0

Available to Guest Users
32.0

Requires Chatter
Yes

Signature
public static ConnectApi.BatchResult[] getUserBatch(String communityId, List<String> userIds)

Parameters
communityId
    Type: String
    ID for an Experience Cloud site, internal, or null.

userIds
    Type: List<String>
    A list of up to 500 user IDs.

Return Value
Type: ConnectApi.BatchResult[]
The ConnectApi.BatchResult.getResult() method returns a ConnectApi.User object and errors for users that didn't load.
Example

```java
// Get users in an organization.
ConnectApi.UserPage userPage = ConnectApi.ChatterUsers.getUsers(null);

// Create a list of user IDs.
List<String> userList = new List<String>();
for (ConnectApi.User user : userPage.users) {
    userList.add(user.id);
}

// Get info about all users in the list.
ConnectApi.BatchResult[] batchResults = ConnectApi.ChatterUsers.getUserBatch(null, userList);
for (ConnectApi.BatchResult batchResult : batchResults) {
    if (batchResult.isSuccess()) {
        // Operation was successful.
        // Print each user's username.
        ConnectApi.UserDetail user;
        if (batchResult.getResult() instanceof ConnectApi.UserDetail) {
            user = (ConnectApi.UserDetail) batchResult.getResult();
        }
        System.debug('SUCCESS');
        System.debug(user.username);
    } else {
        // Operation failed. Print errors.
        System.debug('FAILURE');
        System.debug(batchResult.getErrorMessage());
    }
}
```

getUserGroups(communityId, userId)

Get a user's groups.

API Version

45.0

Available to Guest Users

45.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.UserGroupDetailPage getUserGroups(String communityId, String userId)
```
Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

userId
Type: String
ID for a user.

Return Value
Type: ConnectApi.UserGroupDetailPage

getUserGroups(communityId, userId, pageParam, pageSize)
Get a page of a user’s groups.

API Version
45.0

Available to Guest Users
45.0

Requires Chatter
Yes

Signature
public static ConnectApi.UserGroupDetailPage getUserGroups(String communityId, String userId, Integer pageParam, Integer pageSize)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

userId
Type: String
ID for a user.

pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in \texttt{null}, the default size is 25.

**Return Value**

Type: \texttt{ConnectApi.UserGroupDetailPage}

\textbf{getUsers(communityId)}

Get the first page of users.

**API Version**

28.0

**Available to Guest Users**

32.0

**Requires Chatter**

Yes

**Signature**

\texttt{public static ConnectApi.UserPage getUsers(String communityId)}

**Parameters**

\texttt{communityId}

Type: \texttt{String}

ID for an Experience Cloud site, internal, or \texttt{null}.

**Return Value**

Type: \texttt{ConnectApi.UserPage}

\textbf{getUsers(communityId, pageParam, pageSize)}

Get a page of users.

**API Version**

28.0

**Available to Guest Users**

32.0

**Requires Chatter**

Yes
**getUsers(String communityId, Integer pageParam, Integer pageSize)**

**Parameters**

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- **pageParam**
  - Type: Integer
  - Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

- **pageSize**
  - Type: Integer
  - Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

**Return Value**

Type: ConnectApi.UserPage

**purgeUserActivities(String communityId, String userId)**

Start a job to purge Chatter-related user activity, such as bookmarks, topic endorsements, and votes.

**API Version**

42.0

**Requires Chatter**

Yes

**purgeUserActivities(String communityId, String userId)**

**Parameters**

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- **userId**
  - Type: String
  - ID of the user.
Return Value
Type: `ConnectApi.UserActivitiesJob`

Usage
The following activities can be purged with this method.
- **Bookmark**—User bookmarked a post.
- **ChatterActivity**—Total counts of posts and comments made and likes and comments received for a user.
- **ChatterLike**—User liked a post or comment.
- **CompanyVerify**—User verified comment.
- **DownVote**—User downvoted a post or comment.
- **FeedEntityRead**—User read a post.
- **FeedRead**—User read a feed.
- **Mute**—User muted a post.
- **TopicEndorsement**—User endorsed another user on a topic or received endorsement on a topic.
- **UpVote**—User upvoted a post or comment.

To delete a user's posts and comments, use these methods, respectively.
- `deleteFeedElement(communityId, feedElementId)`
- `deleteComment(communityId, commentId)`

`searchUserGroupDetails(communityId, userId, q)`
Get the user's groups that match the search criteria.

API Version
45.0

Available to Guest Users
45.0

Requires Chatter
Yes

Signature
```java
public static ConnectApi.UserGroupDetailPage searchUserGroupDetails(String communityId, String userId, String q)
```

Parameters
- `communityId` Type: `String`
  ID for an Experience Cloud site, internal, or `null`.

947
userId
Type: String
ID for a user.

q
Type: String
Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

Return Value
Type: ConnectApi.UserGroupDetailPage

searchUserGroupDetails(communityId, userId, q, pageParam, pageSize)
Get a page of a user’s groups that match the search criteria.

API Version
45.0

Available to Guest Users
45.0

Requires Chatter
Yes

Signature
public static ConnectApi.UserGroupDetailPage searchUserGroupDetails(String communityId, String userId, String q, Integer pageParam, Integer pageSize)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

userId
Type: String
ID for a user.

q
Type: String
Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

`pageSize`
Type: `Integer`

Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

Return Value
Type: `ConnectApi.UserGroupDetailPage`

`searchUsers(communityId, q)`
Get the first page of users that match the search criteria.

API Version
28.0

Available to Guest Users
32.0

Requires Chatter
Yes

Signature
`public static ConnectApi.UserPage searchUsers(String communityId, String q)`

Parameters
`communityId`
Type: `String`

ID for an Experience Cloud site, internal, or null.

`q`
Type: `String`

Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

Return Value
Type: `ConnectApi.UserPage`
Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestSearchUsers(communityId, q, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

`searchUsers(communityId, q, pageParam, pageSize)`
Get a page of users that match the search criteria.

API Version
28.0

Available to Guest Users
32.0

Requires Chatter
Yes

Signature
public static ConnectApi.UserPage searchUsers(String communityId, String q, Integer pageParam, Integer pageSize)

Parameters

- `communityId`  
  Type: `String`  
  ID for an Experience Cloud site, `internal`, or `null`.

- `q`  
  Type: `String`  
  Required and can't be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See *Wildcards*.

- `pageParam`  
  Type: `Integer`  
  Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

- `pageSize`  
  Type: `Integer`  
  Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.
Return Value

Type: `ConnectApi.UserPage`

Usage

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- `setTestSearchUsers(communityId, q, pageParam, pageSize, result)

`searchUsers(communityId, q, searchContextId, pageParam, pageSize)`

Get a page of users that match the search criteria.

API Version

28.0

Available to Guest Users

32.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.UserPage searchUsers(String communityId, String q, String searchContextId, Integer pageParam, Integer pageSize)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- `q`
  - Type: `String`
  - Required and can't be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See `Wildcards`.

- `searchContextId`
  - Type: `String`
  - A feed item ID that filters search results for feed @mentions. More useful results are listed first. When you specify this argument, you cannot query more than 500 results and you cannot use wildcards in the search term.
pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

Return Value
Type: ConnectApi.UserPage

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestSearchUsers(communityId, q, searchContextId, pageParam, pageSize, result)

updateChatterSettings(communityId, userId, defaultGroupEmailFrequency)
Update the default Chatter settings for a user.

API Version
28.0

Requires Chatter
Yes

Signature
public static ConnectApi.UserChatterSettings updateChatterSettings(String communityId, String userId, ConnectApi.GroupEmailFrequency defaultGroupEmailFrequency)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

userId
Type: String
ID for the context user or the keyword me.

defaultGroupEmailFrequency
Type: ConnectApi.GroupEmailFrequency
Frequency with which a user receives email. Values are:

- EachPost
- DailyDigest
- WeeklyDigest
- Never
- UseDefault

Don’t pass the value UseDefault for the `defaultGroupEmailFrequency` parameter because calling `updateChatterSettings` sets the default value.

Return Value

Type: `ConnectApi.UserChatterSettings`

`updateUser(communityId, userId, userInput)`

Update the About Me section for a user.

API Version

29.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.UserDetail updateUser(String communityId, String userId, ConnectApi.UserInput userInput)
```

Parameters

- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.

- `userId`
  Type: `String`
  ID for the context user or the keyword `me`.

- `userInput`
  Type: `ConnectApi.UserInput`
  Specifies the updated information.

Return Value

Type: `ConnectApi.UserDetail`
ChatterUsers Test Methods

The following are the test methods for ChatterUsers. All methods are static.
For information about using these methods to test your ConnectApi code, see Testing ConnectApi Code.

IN THIS SECTION:

setTestSearchUsers(communityId, q, result)
Register a ConnectApi.UserPage object to be returned when the matching ConnectApi.searchUsers method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchUsers(communityId, q, pageParam, pageSize, result)
Register a ConnectApi.UserPage object to be returned when the matching ConnectApi.searchUsers method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchUsers(communityId, q, searchContextId, pageParam, pageSize, result)
Register a ConnectApi.UserPage object to be returned when the matching ConnectApi.searchUsers method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchUsers(communityId, q, result)
Register a ConnectApi.UserPage object to be returned when the matching ConnectApi.searchUsers method is called in a test context. Use the method with the same parameters or you receive an exception.

setTestSearchUsers(communityId, q, result)
Register a ConnectApi.UserPage object to be returned when the matching ConnectApi.searchUsers method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
28.0

Signature

public static Void setTestSearchUsers(String communityId, String q, ConnectApi.UserPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

result
Type: ConnectApi.UserPage
Object containing test data.
Return Value
Type: Void

SEE ALSO:
searchUsers(communityId, q)

setTestSearchUsers(communityId, q, pageParam, pageSize, result)
Register a ConnectApi.UserPage object to be returned when the matching ConnectApi.searchUsers method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
28.0

Signature
public static Void setTestSearchUsers(String communityId, String q, Integer pageParam, Integer pageSize, ConnectApi.UserPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
Required and can't be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

result
Type: ConnectApi.UserPage
Object containing test data.
Return Value
Type: Void

SEE ALSO:
searchUsers(communityId, q, pageParam, pageSize)

**setTestSearchUsers(communityId, q, searchContextId, pageParam, pageSize, result)**

Register a `ConnectApi.UserPage` object to be returned when the matching `ConnectApi.searchUsers` method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**
28.0

**Signature**
```
public static Void setTestSearchUsers(String communityId, String q, String searchContextId, Integer pageParam, Integer pageSize, ConnectApi.UserPage result)
```

**Parameters**

*communityId*
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

*q*
Type: `String`
Required and can’t be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

*searchContextId*
Type: `String`
A feed item ID that filters search results for feed @mentions. More useful results are listed first. When you specify this argument, you cannot query more than 500 results and you cannot use wildcards in the search term.

*pageParam*
Type: `Integer`
Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

*pageSize*
Type: `Integer`
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

*result*
Type: `ConnectApi.UserPage`
Object containing test data.
Return Value

Type: Void

SEE ALSO:

searchUsers(communityId, q, searchContextId, pageParam, pageSize)


Retired ChatterUsers Methods

The following methods for ChatterUsers are retired.

IN THIS SECTION:

deletePhoto(communityId, userId)
Delete a user's photo.

groups(communityId, userId)
Get the groups that a user is a member of.

groups(communityId, userId, pageParam, pageSize)
Get a page of groups that a user is a member of.

getPhoto(communityId, userId)
Get a user's photo.

searchUserGroups(communityId, userId, q)
Get the user's groups that match the search criteria.

searchUserGroups(communityId, userId, q, pageParam, pageSize)
Get a page of a user's groups that match the search criteria.

setPhoto(communityId, userId, fileId, versionNumber)
Set an uploaded file as a user's photo.

setPhoto(communityId, userId, fileUpload)
Set a file that hasn't been uploaded as the user's photo.

setPhotoWithAttributes(communityId, userId, photo)
Set and crop an uploaded file as a user's photo.

setPhotoWithAttributes(communityId, userId, photo, fileUpload)
Set and crop a file that hasn't been uploaded as a user's photo.


deletePhoto(communityId, userId)
Delete a user's photo.

API Version

28.0–34.0

Important: In version 35.0 and later, use ConnectApi.UserProfiles.deletePhoto(communityId, userId)
Signature

```
public static Void deletePhoto(String communityId, String userId)
```

Parameters

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- **userId**
  - Type: String
  - ID for the context user or the keyword me.

Return Value

Type: Void

**getGroups(communityId, userId)**

Get the groups that a user is a member of.

API Version

28.0–44.0

⚠️ Important: In version 45.0 and later, use `getUserGroups(communityId, userId)`.

Available to Guest Users

32.0–44.0

Requires Chatter

Yes

Signature

```
public static ConnectApi.UserGroupPage getGroups(String communityId, String userId)
```

Parameters

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- **userId**
  - Type: String
  - ID for a user.
Return Value
Type: `ConnectApi.UserGroupPage`

`getGroups(communityId, userId, pageParam, pageSize)`
Get a page of groups that a user is a member of.

API Version
28.0–44.0

⚠️ Important: In version 45.0 and later, use `getUserGroups(communityId, userId, pageParam, pageSize)`.

Available to Guest Users
32.0–44.0

Requires Chatter
Yes

Signature

```java
public static ConnectApi.UserGroupPage getGroups(String communityId, String userId,
    Integer pageParam, Integer pageSize)
```

Parameters

- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.
- `userId`
  Type: `String`
  ID for a user.
- `pageParam`
  Type: `Integer`
  Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.
- `pageSize`
  Type: `Integer`
  Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

Return Value
Type: `ConnectApi.UserGroupPage`

`getPhoto(communityId, userId)`
Get a user’s photo.
API Version
28.0–34.0

**Important:** In version 35.0 and later, use `ConnectApi.UserProfiles.getPhoto(communityId, userId)`.

Available to Guest Users
32.0

Requires Chatter
Yes

Signature

```java
public static ConnectApi.Photo getPhoto(String communityId, String userId)
```

Parameters

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.
- **userId**
  - Type: `String`
  - ID for a user.

Return Value

Type: `ConnectApi.Photo`

`searchUserGroups(communityId, userId, q)`

Get the user’s groups that match the search criteria.

API Version

30.0–44.0

**Important:** In version 45.0 and later, use `searchUserGroupDetails(communityId, userId, q)`.

Available to Guest Users

32.0–44.0

Requires Chatter

Yes
Signature

```java
public static ConnectApi.UserGroupPage searchUserGroups(String communityId, String userId, String q)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- `userId`
  - Type: `String`
  - ID for a user.

- `q`
  - Type: `String`
  - Required and can't be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

Return Value

Type: `ConnectApi.UserGroupPage`

```java
searchUserGroups(communityId, userId, q, pageParam, pageSize)
```

Get a page of a user’s groups that match the search criteria.

API Version

30.0–44.0

**Important:** In version 45.0 and later, use `searchUserGroupDetails(communityId, userId, q, pageParam, pageSize)`.

Available to Guest Users

32.0–44.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.UserGroupPage searchUserGroups(String communityId, String userId, String q, Integer pageParam, Integer pageSize)
```

Parameters

- `communityId`
  - Type: `String`
ID for an Experience Cloud site, internal, or null.

**userId**
- Type: String
- ID for a user.

**q**
- Type: String
- Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

**pageParam**
- Type: Integer
- Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

**pageSize**
- Type: Integer
- Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

**Return Value**
- Type: ConnectApi.UserGroupPage

**setPhoto(communityId, userId, fileId, versionNumber)**
Set an uploaded file as a user’s photo.

**API Version**
- 28.0–34.0

⚠️ **Important:** In version 35.0 and later, use `ConnectApi.UserProfiles.setPhoto(communityId, userId, fileId, versionNumber)`

**Requires Chatter**
- Yes

**Signature**
- `public static ConnectApi.Photo setPhoto(String communityId, String userId, String fileId, Integer versionNumber)`

**Parameters**

**communityId**
- Type: String
- ID for an Experience Cloud site, internal, or null.

**userId**
- Type: String
- ID for the context user or the keyword me.
fileId
  Type: String
  ID of a file already uploaded. The file must be an image, and be smaller than 2 GB.

versionNumber
  Type: Integer
  Version number of the existing file. Specify either an existing version number, or null to get the latest version.

Return Value
Type: ConnectApi.Photo

Usage
Photos are processed asynchronously and might not be visible right away.

setPhoto(communityId, userId, fileUpload)
Set a file that hasn’t been uploaded as the user’s photo.

API Version
28.0–34.0

⚠️ Important: In version 35.0 and later, use ConnectApi.UserProfiles.setPhoto(communityId, userId, fileUpload)

Requires Chatter
Yes

Signature
public static ConnectApi.Photo setPhoto(String communityId, String userId, ConnectApi.BinaryInput fileUpload)

Parameters
communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

userId
  Type: String
  ID for the context user or the keyword me.

fileUpload
  Type: ConnectApi.BinaryInput
  File to use as the photo. The content type must be usable as an image.
Return Value
Type: ConnectApi.Photo

Usage
Photos are processed asynchronously and might not be visible right away.

**setPhotoWithAttributes**(communityId, userId, photo)
Set and crop an uploaded file as a user’s photo.

API Version
29.0–34.0

⚠️ **Important**: In version 35.0 and later, use
`ConnectApi.UserProfiles.setPhotoWithAttributes(communityId, userId, photo)`

Requires Chatter
Yes

Signature
`public static ConnectApi.Photo setPhotoWithAttributes(String communityId, String userId,
ConnectApi.PhotoInput photo)`

Parameters
- **communityId**
  Type: String
  ID for an Experience Cloud site, internal, or null.
- **userId**
  Type: String
  ID for the context user or the keyword me.
- **photo**
  Type: ConnectApi.PhotoInput
  A ConnectApi.PhotoInput object specifying the file ID, version number, and cropping parameters.

Return Value
Type: ConnectApi.Photo

Usage
Photos are processed asynchronously and might not be visible right away.
**setPhotoWithAttributes**(communityId, userId, photo, fileUpload)

Set and crop a file that hasn’t been uploaded as a user’s photo.

**API Version**

29.0–34.0

**Important:** In version 35.0 and later, use

```java
ConnectApi.UserProfiles.setPhotoWithAttributes(communityId, userId, photo, fileUpload)
```

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.Photo setPhotoWithAttributes(String communityId, String userId,
ConnectApi.PhotoInput photo, ConnectApi.BinaryInput fileUpload)
```

**Parameters**

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **userId**
  - Type: `String`
  - ID for the context user or the keyword `me`.

- **photo**
  - Type: `ConnectApi.PhotoInput`
  - A `ConnectApi.PhotoInput` object specifying the cropping parameters.

- **fileUpload**
  - Type: `ConnectApi.BinaryInput`
  - File to use as the photo. The content type must be usable as an image.

**Return Value**

Type: `ConnectApi.Photo`

**Usage**

Photos are processed asynchronously and might not be visible right away.

**Clm Class**

Create and update Contract Lifecycle Management (CLM) contracts using object ID.
 Namespace

ConnectApi

Clm Methods

The following are methods for Clm. All methods are static.

IN THIS SECTION:

createContract(contractInputPayload)
Create contracts using the object ID.

updateContract(contractInputPayload)
Update contracts using the object ID.

createContract(contractInputPayload)
Create contracts using the object ID.

API Version
56.0

Requires Chatter
No

Signature


Parameters

contractInputPayload
Type: ConnectApi.ContractInputRepresentation on page 1578
Input payload to create contract.

Return Value

Type: ConnectApi.ContractOutputRepresentation on page 1766

updateContract(contractInputPayload)
Update contracts using the object ID.

API Version
56.0
Requires Chatter
No

Signature
public static ConnectApi.ContractOutputRepresentation
updateContract(ConnectApi.ContractInputRepresentation contractInputPayload)

Parameters
contractInputPayload
Type: ConnectApi.ContractInputRepresentation on page 1578
Input payload to update contract.

Return Value
Type: ConnectApi.ContractOutputRepresentation on page 1766

CommerceBuyerExperience Class
Create, delete, or get commerce addresses. Get order delivery group, order item, order shipments, shipment items, and order summaries. Get adjustments for order items and order summaries.

Namespace
ConnectApi

CommerceBuyerExperience Methods
These methods are for CommerceBuyerExperience. All methods are static.

IN THIS SECTION:
addOrderToCart(webstoreId, orderSummaryId, orderToCartInput)
Add an order to a cart using a webstore order summary.
addOrderToCart(webstoreId, orderSummaryId, orderToCartInput, effectiveAccountId)
Add an order to a cart for a specific account using a webstore order summary.
calculateAdjustmentAggregates(webstoreId, orderSummaryIds)
Submit a job to calculate adjustment aggregates for a list of order summary IDs.
createCommerceAccountAddress(webstoreId, accountld, addressInput)
Create a Commerce account address for a webstore account.
deleteCommerceAccountAddress(webstoreId, accountld, addressId)
Delete a Commerce account address for a webstore.
getCommerceAccountAddress(webstoreId, accountld)
Get a Commerce account address for a webstore.
getCommerceAccountAddress(webstoreId, accountId, defaultOnly)
Get Commerce account addresses for a webstore and account.

getCommerceAccountAddress(webstoreId, accountId, defaultOnly, addressType, fields, pageToken, pageSize, sortOrder)
Get Commerce account addresses for a webstore and account.

getCommerceAccountAddress(webstoreId, accountId, defaultOnly, addressType, excludeUnsupportedCountries)
Get Commerce account addresses for a webstore and account.

getCommerceAccountAddress(webstoreId, accountId, defaultOnly, addressType, excludeUnsupportedCountries, fields, pageToken, pageSize, sortOrder)
Get Commerce account addresses for a webstore and account.

getOrderDeliveryGroupSummaries(webstoreId, effectiveAccountId, orderSummaryId)
Get order delivery group summaries.

getOrderDeliveryGroupSummaries(webstoreId, effectiveAccountId, orderSummaryId, pageSize)
Get order delivery group summaries.

getOrderDeliveryGroupSummaries(webstoreId, effectiveAccountId, orderSummaryId, pageParam)
Get a page of order delivery group summaries.

getOrderDeliveryGroupSummaries(webstoreId, effectiveAccountId, orderSummaryId, fields)
Get order delivery group summaries with specific fields.

getOrderDeliveryGroupSummaries(webstoreId, effectiveAccountId, orderSummaryId, pageParam, fields)
Get a page of order delivery group summaries with specific fields.

getOrderDeliveryGroupSummaries(webstoreId, effectiveAccountId, orderSummaryId, fields, pageSize)
Get order delivery group summaries with specific fields.

getOrderDeliveryGroupSummaries(webstoreId, effectiveAccountId, orderSummaryId, fields, sortParam)
Get a sorted list of order delivery group summaries with specific fields.

getOrderDeliveryGroupSummaries(webstoreId, effectiveAccountId, orderSummaryId, fields, pageSize, sortParam)
Get a sorted list of order delivery group summaries with specific fields.

getAddressSummaries(webstoreId, effectiveAccountId, orderSummaryId)
Get order item summaries.

getAddressSummaries(webstoreId, effectiveAccountId, orderSummaryId, pageSize)
Get order item summaries.

getAddressSummaries(webstoreId, effectiveAccountId, orderSummaryId, orderDeliveryGroupSummaryId)
Get order item summaries for a delivery group summary.

getAddressSummaries(webstoreId, effectiveAccountId, orderSummaryId, orderDeliveryGroupSummaryId, pageSize)
Get order item summaries for a delivery group summary.

getAddressSummaries(webstoreId, effectiveAccountId, orderSummaryId, orderDeliveryGroupSummaryId, pageParam)
Get a page of order item summaries for a delivery group summary.

getAddressSummaries(webstoreId, effectiveAccountId, orderSummaryId, orderDeliveryGroupSummaryId, fields)
Get order item summaries for a delivery group summary with specific fields.

getAddressSummaries(webstoreId, effectiveAccountId, orderSummaryId, orderDeliveryGroupSummaryId, fields, pageSize)
Get order item summaries for a delivery group summary with specific fields.
getOrderItemSummaries(webstoreId, effectiveAccountId, orderSummaryId, orderDeliveryGroupSummaryId, fields, pageParam)
Get a page of order item summaries for a delivery group summary with specific fields.

getoOrderItemSummaries(webstoreId, effectiveAccountId, orderSummaryId, orderDeliveryGroupSummaryId, fields, sortParam)
Get a sorted list of order item summaries for a delivery group summary with specific fields.

getoOrderItemSummaries(webstoreId, effectiveAccountId, orderSummaryId, orderDeliveryGroupSummaryId, fields, pageSize, sortParam)
Get a sorted page of order item summaries for a delivery group summary with specific fields.

getoOrderItemSummaries(webstoreId, effectiveAccountId, orderSummaryId, orderDeliveryGroupSummaryId, fields, pageParam, pageSize, sortParam, includeAdjustmentDetails)
Get a sorted page of order item summaries for a delivery group summary with specific fields and include adjustment details.

getoOrderItemSummaryAdjustments(webstoreId, orderSummaryId, orderItemSummaryAdjustmentCollectionInput)
Get adjustments for order items.

getoOrderItemSummaryAdjustments(webstoreId, orderSummaryId, orderItemSummaryAdjustmentCollectionInput, effectiveAccountId)
Get adjustments for order items.

getoOrderShipmentItems(webstoreId, shipmentId)
Get order shipment items.

getoOrderShipmentItems(webstoreId, shipmentId, effectiveAccountId)
Get order shipment items.

getoOrderShipmentItems(webstoreId, shipmentId, effectiveAccountId, fields)
Get order shipment items with specific fields.

getoOrderShipmentItems(webstoreId, shipmentId, effectiveAccountId, fields, pageToken, pageSize)
Get a page of order shipment items with specific fields.

getoOrderShipmentItems(webstoreId, shipmentId, effectiveAccountId, fields, pageToken, pageSize, sortOrder)
Get a sorted page of order shipment items.

getoOrderShipments(webstoreId, orderSummaryId)
Get order shipments.

getoOrderShipments(webstoreId, orderSummaryId, effectiveAccountId)
Get order shipments.

getoOrderShipments(webstoreId, orderSummaryId, effectiveAccountId, fields)
Get order shipments with specific fields.

getoOrderShipments(webstoreId, orderSummaryId, effectiveAccountId, fields, pageSize, pageToken)
Get a page of order shipments with specific fields.

getoOrderShipments(webstoreId, orderSummaryId, effectiveAccountId, fields, pageSize, pageToken, sortOrder)
Get a sorted page of order shipments with specific fields.

getoOrderSummaries(webstoreId)
Get order summaries.

getoOrderSummaries(webstoreId, effectiveAccountId)
Get order summaries.

getoOrderSummaries(webstoreId, effectiveAccountId, fields)
Get order summaries with specific fields.

getoOrderSummaries(webstoreId, effectiveAccountId, fields, pageSize, pageToken)
Get a page of order summaries with specific fields.
getOrderSummaries(webstoreId, effectiveAccountId, fields, pageSize, pageToken, sortOrder)
Get a sorted page of order summaries with specific fields.

getOrderSummaries(webstoreId, effectiveAccountId, fields, pageSize, pageToken, sortOrder, earliestDate, latestDate)
Get a sorted page of order summaries with specific fields within a specific date range.

getOrderSummaries(webstoreId, effectiveAccountId, fields, pageSize, pageToken, sortOrder, earliestDate, latestDate, ownerScoped)
Get a sorted page of order summaries with specific fields within a specific date range and scoped to orders owned by the context user.

getOrderSummaries(webstoreId, effectiveAccountId, fields, pageSize, pageToken, sortOrder, earliestDate, latestDate, ownerScoped, includeAdjustmentDetails)
Get a sorted page of order summaries with specific fields within a specific date range and scoped to orders owned by the context user.

g.getOrderSummary(webstoreId, orderSummaryId, effectiveAccountId)
Get an order summary.

g.getOrderSummary(webstoreId, orderSummaryId, effectiveAccountId, fields)
Get an order summary with fields.

g.getOrderSummary(webstoreId, orderSummaryId, effectiveAccountId, fields, includeAdjustmentDetails)
Get an order summary with fields and include adjustment details.

g.getOrderSummaryAdjustments(webstoreId, orderSummaryId)
Get adjustments for an order summary.

g.getOrderSummaryAdjustments(webstoreId, orderSummaryId, effectiveAccountId)
Get adjustments for an order summary.

lookupOrderSummary(webstoreId, effectiveAccountId, fields, excludeLineItems, excludeDeliveryGroups, excludeAdjustmentAggregates, excludeAdjustments, deliveryGroupId, orderSummaryLookupInput) (Developer Preview)
Look up details about an order summary for a guest shopper or a registered buyer using the effective account ID, requested fields, line items, delivery groups, adjustments aggregates, and adjustments.

lookupOrderSummary(webstoreId, effectiveAccountId, fields, orderSummaryLookupInput) (Developer Preview)
Look up details about an order summary for a guest shopper or a registered buyer using the effective account ID and requested fields.

lookupOrderSummary(webstoreId, effectiveAccountId, orderSummaryLookupInput) (Developer Preview)
Look up details about an order summary for a guest shopper or a registered buyer using the effective account ID.

updateCommerceAccountAddress(webstoreId, accountId, addressId, addressInput)
Update a Commerce account address for a webstore.

addOrderToCart(webstoreId, orderSummaryId, orderToCartInput)
Add an order to a cart using a webstore order summary.

**API Version**

57.0

**Requires Chatter**

No
Signature

```java
public static ConnectApi.OrderToCartResult addOrderToCart(String webstoreId, String orderSummaryId, ConnectApi.OrderToCartInput orderToCartInput)
```

Parameters

- `webstoreId`
  - Type: `String`
  - ID of the webstore.
- `orderSummaryId`
  - Type: `String`
  - ID of the order summary.
- `orderToCartInput`
  - Type: `ConnectApi.OrderToCartInput`
  - Input value indicating which cart the order should be added to.

Return Value

- Type: `ConnectApi.OrderToCartResult`

```
addOrderToCart(webstoreId, orderSummaryId, orderToCartInput, effectiveAccountId)
```

Add an order to a cart for a specific account using a webstore order summary.

API Version

57.0

Requires Chatter

No

Signature

```java
public static ConnectApi.OrderToCartResult addOrderToCart(String webstoreId, String orderSummaryId, ConnectApi.OrderToCartInput orderToCartInput, String effectiveAccountId)
```

Parameters

- `webstoreId`
  - Type: `String`
  - ID of the webstore.
- `orderSummaryId`
  - Type: `String`
  - ID of the order summary.
orderToCartInput
   Type: ConnectApi.OrderToCartInput
   Input value indicating which cart the order should be added to.

effectiveAccountId
   Type: String
   ID of the account for which the request is made. If null, defaults to the account ID for the context user.

Return Value
   Type: ConnectApi.OrderToCartResult

calculateAdjustmentAggregates(webstoreId, orderSummaryIds)
Submit a job to calculate adjustment aggregates for a list of order summary IDs.

API Version
55.0

Requires Chatter
No

Signature
public static ConnectApi.OrderSummaryAdjustmentAggregatesAsyncOutput calculateAdjustmentAggregates(String webstoreId,
ConnectApi.OrderSummaryAdjustmentAggregatesAsyncInput orderSummaryIds)

Parameters
webstoreId
   Type: String
   ID of the webstore.

orderSummaryIds
   Type: ConnectApi.OrderSummaryAdjustmentAggregatesAsyncInput
   A ConnectApi.OrderSummaryAdjustmentAggregatesAsyncInput class with a list of order summary IDs.

Return Value
   Type: ConnectApi.OrderSummaryAdjustmentAggregatesAsyncOutput

createCommerceAccountAddress(webstoreId, accountId, addressInput)
Create a Commerce account address for a webstore account.
API Version
54.0

Requires Chatter
No

Signature
public static ConnectApi.CommerceAddressOutput createCommerceAccountAddress(String webstoreId, String accountId, ConnectApi.commerceAddressInput addressInput)

Parameters
webstoreId
Type: String
ID of the webstore.

accountId
Type: String
ID of the account.

addressInput
Type: ConnectApi.commerceAddressInput
Information about the address you want to create.

Return Value
Type: ConnectApi.CommerceAddressOutput

deleteCommerceAccountAddress(webstoreId, accountId, addressId)
Delete a Commerce account address for a webstore.

API Version
54.0

Requires Chatter
No

Signature
public static Void deleteCommerceAccountAddress(String webstoreId, String accountId, String addressId)
Parameters

**webstoreId**
Type: String
ID of the webstore.

**accountId**
Type: String
ID of the account.

**addressId**
Type: String
ID of the address.

Return Value
Type: Void

**getCommerceAccountAddress(webstoreId, accountId)**
Get a Commerce account address for a webstore.

API Version
54.0

Requires Chatter
No

Signature

```java
public static ConnectApi.CommerceAddressCollection getCommerceAccountAddress(String webstoreId, String accountId)
```

Parameters

**webstoreId**
Type: String
ID of the webstore.

**accountId**
Type: String
ID of the account.

Return Value
Type: `ConnectApi.CommerceAddressCollection`
**getCommerceAccountAddress** *(webstoreId, accountId, defaultOnly)*

Get Commerce account addresses for a webstore and account.

You can get the default address by itself, or you can get all of the addresses for the account.

**API Version**

54.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.CommerceAddressCollection getCommerceAccountAddress(String webstoreId, String accountId, Boolean defaultOnly)
```

**Parameters**

- **webstoreId**
  - Type: String
  - ID of the webstore.

- **accountId**
  - Type: String
  - ID of the account.

- **defaultOnly**
  - Type: Boolean
  - Indicate if you only want the default address (true) or all addresses for the account (false). The default value is false.

**Return Value**

Type: `ConnectApi.CommerceAddressCollection`

---

**getCommerceAccountAddress** *(webstoreId, accountId, defaultOnly, addressType, fields, pageToken, pageSize, sortOrder)*

Get Commerce account addresses for a webstore and account.

**API Version**

54.0

**Requires Chatter**

No
Signature

```java
public static ConnectApi.CommerceAddressCollection getCommerceAccountAddress(String webstoreId, String accountId, Boolean defaultOnly, List<String> addressType, List<String> fields, String pageToken, Integer pageSize, ConnectApi.CommerceAddressSort sortOrder)
```

Parameters

- **webstoreId**
  - Type: String
  - ID of the webstore.

- **accountId**
  - Type: String
  - ID of the account.

- **defaultOnly**
  - Type: Boolean
  - Indicate if you want only the default address (true) or all addresses for the account (false). The default value is false.

- **addressType**
  - Type: List<String>
  - Type of address, for example, Billing or Shipping.

- **fields**
  - Type: List<String>
  - A list of custom fields for the address.

- **pageToken**
  - Type: String
  - Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in null, the first page is returned.

- **pageSize**
  - Type: Integer
  - Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

- **sortOrder**
  - Type: ConnectApi.CommerceAddressSort
  - Sort order for Commerce addresses.
  - - CreatedDateAsc—Sort in ascending order of created date.
  - - CreatedDateDesc—Sort in descending order of created date.
  - - NameAsc—Sort in ascending order of name.
  - - NameDesc—Sort in descending order of name.

Return Value

- Type: ConnectApi.CommerceAddressCollection
getCommerceAccountAddress(webstoreId, accountId, addressType, excludeUnsupportedCountries)

Get Commerce account addresses for a webstore and account.

API Version
57.0

Requires Chatter
No

Signature

public static ConnectApi.CommerceAddressCollection getCommerceAccountAddress(String webstoreId, String accountId, List<String> addressType, Boolean excludeUnsupportedCountries)

Parameters

webstoreId
Type: String
ID of the webstore.

accountId
Type: String
ID of the account.

addressType
Type: List<String>
Type of address, for example, Billing or Shipping.

excludeUnsupportedCountries
Type: Boolean
Indicate if you want to retrieve all addresses (false) or only addresses of type Shipping that are in countries included in the store’s shipToCountries list (true). The default value is false.

Return Value
Type: ConnectApi.CommerceAddressCollection

getCommerceAccountAddress(webstoreId, accountId, defaultOnly, addressType, excludeUnsupportedCountries)

Get Commerce account addresses for a webstore and account.

API Version
57.0
Requires Chatter
No

Signature

public static ConnectApi.CommerceAddressCollection getCommerceAccountAddress(String webstoreId, String accountId, Boolean defaultOnly, List<String> addressType, Boolean excludeUnsupportedCountries)

Parameters

webstoreId
Type: String
ID of the webstore.

accountId
Type: String
ID of the account.

defaultOnly
Type: Boolean
Indicate if you want only the default address (true) or all addresses for the account (false). The default value is false.

addressType
Type: List<String>
Type of address, for example, Billing or Shipping.

excludeUnsupportedCountries
Type: Boolean
Indicate if you want to retrieve all addresses (false) or only addresses of type Shipping that are in countries included in the store’s shipToCountries list (true). The default value is false.

Return Value

Type: ConnectApi.CommerceAddressCollection

getCommerceAccountAddress(webstoreId, accountId, defaultOnly, addressType, excludeUnsupportedCountries, fields, pageToken, pageSize, sortOrder)

Get Commerce account addresses for a webstore and account.

API Version

57.0

Requires Chatter
No
public static ConnectApi.CommerceAddressCollection getCommerceAccountAddress(String webstoreId, String accountId, Boolean defaultOnly, List<String> addressType, Boolean excludeUnsupportedCountries, List<String> fields, String pageToken, Integer pageSize, ConnectApi.CommerceAddressSort sortOrder)

Parameters

webstoreId
Type: String
ID of the webstore.

accountId
Type: String
ID of the account.

defaultOnly
Type: Boolean
Indicate if you want only the default address (true) or all addresses for the account (false). The default value is false.

addressType
Type: List<String>
Type of address, for example, Billing or Shipping.

excludeUnsupportedCountries
Type: Boolean
Indicate if you want to retrieve all addresses (false) or only addresses of type Shipping that are in countries included in the store’s shipToCountries list (true). The default value is false.

fields
Type: List<String>
A list of custom fields for the address.

pageToken
Type: String
Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortOrder
Type: ConnectApi.CommerceAddressSort
Sort order for Commerce addresses. Values are:
- CreatedDateAsc—Sort in ascending order of created date.
- CreatedDateDesc—Sort in descending order of created date.
- NameAsc—Sort in ascending order of name.
- NameDesc—Sort in descending order of name.
Return Value
Type: `ConnectApi.CommerceAddressCollection`

`getOrderDeliveryGroupSummaries(webstoreId, effectiveAccountId, orderSummaryId)`
Get order delivery group summaries.

API Version
51.0

Requires Chatter
No

Signature

```java
public static ConnectApi.OrderDeliveryGroupSummaryCollection
getOrderDeliveryGroupSummaries(String webstoreId, String effectiveAccountId, String orderSummaryId)
```

Parameters

- `webstoreId`
  Type: `String`
  ID of the webstore.
- `effectiveAccountId`
  Type: `String`
  ID of the account for which the request is made. If `null`, defaults to the account ID for the context user.
- `orderSummaryId`
  Type: `String`
  ID of the order summary.

Return Value
Type: `ConnectApi.OrderDeliveryGroupSummaryCollection`

`getOrderDeliveryGroupSummaries(webstoreId, effectiveAccountId, orderSummaryId, pageSize)`
Get order delivery group summaries.

API Version
51.0
Requires Chatter
No

Signature

```java
public static ConnectApi.OrderDeliveryGroupSummaryCollection
getOrderDeliveryGroupSummaries(String webstoreId, String effectiveAccountId, String orderSummaryId, Integer pageSize)
```

Parameters

- `webstoreId`
  - Type: `String`
  - ID of the webstore.

- `effectiveAccountId`
  - Type: `String`
  - ID of the account for which the request is made. If `null`, defaults to the account ID for the context user.

- `orderSummaryId`
  - Type: `String`
  - ID of the order summary.

- `pageSize`
  - Type: `Integer`
  - Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

Return Value

Type: `ConnectApi.OrderDeliveryGroupSummaryCollection`

```java
getOrderDeliveryGroupSummaries(webstoreId, effectiveAccountId, orderSummaryId, pageParam)
```

Get a page of order delivery group summaries.

API Version

51.0

Requires Chatter
No

Signature

```java
public static ConnectApi.OrderDeliveryGroupSummaryCollection
getOrderDeliveryGroupSummaries(String webstoreId, String effectiveAccountId, String orderSummaryId, String pageParam)
```
Parameters

webstoreId
Type: String
ID of the webstore.

effectiveAccountId
Type: String
ID of the account for which the request is made. If null, defaults to the account ID for the context user.

orderSummaryId
Type: String
ID of the order summary.

pageParam
Type: String
Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in null, the first page is returned.

Return Value
Type: ConnectApi.OrderDeliveryGroupSummaryCollection

getOrderDeliveryGroupSummaries(webstoreId, effectiveAccountId, orderSummaryId, fields)
Get order delivery group summaries with specific fields.

API Version
51.0

Requires Chatter
No

Signature
public static ConnectApi.OrderDeliveryGroupSummaryCollection
getOrderDeliveryGroupSummaries(String webstoreId, String effectiveAccountId, String orderSummaryId, List<String> fields)

Parameters

webstoreId
Type: String
ID of the webstore.

effectiveAccountId
Type: String
ID of the account for which the request is made. If null, defaults to the account ID for the context user.
orderSummaryId
   Type: String
   ID of the order summary.

fields
   Type: List<String>
   List of up to 15 order delivery group summary or order delivery method fields to display in the UI in each item row.

Return Value
Type: ConnectApi.OrderDeliveryGroupSummaryCollection

getOrderDeliveryGroupSummaries(webstoreId, effectiveAccountId, orderSummaryId, pageParam, fields)
Get a page of order delivery group summaries with specific fields.

API Version
51.0

Requires Chatter
No

Signature
public static ConnectApi.OrderDeliveryGroupSummaryCollection
getOrderDeliveryGroupSummaries(String webstoreId, String effectiveAccountId, String orderSummaryId, String pageParam, List<String> fields)

Parameters
webstoreId
   Type: String
   ID of the webstore.

effectiveAccountId
   Type: String
   ID of the account for which the request is made. If null, defaults to the account ID for the context user.

orderSummaryId
   Type: String
   ID of the order summary.

pageParam
   Type: String
   Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in null, the first page is returned.
**Fields**

Type: `List<String>`

List of up to 15 order delivery group summary or order delivery method fields to display in the UI in each item row.

**Return Value**

Type: `ConnectApi.OrderDeliveryGroupSummaryCollection`

**getOrderDeliveryGroupSummaries(webstoreId, effectiveAccountId, orderSummaryId, fields, pageSize)**

Get order delivery group summaries with specific fields.

**API Version**

51.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.OrderDeliveryGroupSummaryCollection
    getOrderDeliveryGroupSummaries(String webstoreId, String effectiveAccountId, String orderSummaryId, List<String> fields, Integer pageSize)
```

**Parameters**

- `webstoreId`
  
  **Type:** `String`
  
  ID of the webstore.

- `effectiveAccountId`
  
  **Type:** `String`
  
  ID of the account for which the request is made. If `null`, defaults to the account ID for the context user.

- `orderSummaryId`
  
  **Type:** `String`
  
  ID of the order summary.

- `fields`
  
  **Type:** `List<String>`
  
  List of up to 15 order delivery group summary or order delivery method fields to display in the UI in each item row.

- `pageSize`
  
  **Type:** `Integer`
  
  Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.
Return Value
Type: `ConnectApi.OrderDeliveryGroupSummaryCollection`

`getOrderDeliveryGroupSummaries(webstoreId, effectiveAccountId, orderSummaryId, fields, sortParam)`
Get a sorted list of order delivery group summaries with specific fields.

API Version
S1.0

Requires Chatter
No

Signature
```java
public static ConnectApi.OrderDeliveryGroupSummaryCollection
getOrderDeliveryGroupSummaries(String webstoreId, String effectiveAccountId, String orderSummaryId, List<String> fields, ConnectApi.OrderDeliveryGroupSummarySort sortParam)
```

Parameters
- **webstoreId**
  - Type: `String`
  - ID of the webstore.
- **effectiveAccountId**
  - Type: `String`
  - ID of the account for which the request is made. If `null`, defaults to the account ID for the context user.
- **orderSummaryId**
  - Type: `String`
  - ID of the order summary.
- **fields**
  - Type: `List<String>`
  - List of up to 15 order delivery group summary or order delivery method fields to display in the UI in each item row.
- **sortParam**
  - Type: `ConnectApi.OrderDeliveryGroupSummarySort`
  - Sort order for order delivery group summaries. Values are:
    - `IdAsc`—Sorts by ID in ascending alphanumeric order (A–Z, 0–9).
    - `IdDesc`—Sorts by ID in descending alphanumeric order (Z–A, 9–0).
  - If `null`, `IdAsc` is the default sort order.
Return Value
Type: ConnectApi.OrderDeliveryGroupSummaryCollection

getOrderDeliveryGroupSummaries(webstoreId, effectiveAccountId, orderSummaryId, fields, pageSize, sortParam)
Get a sorted list of order delivery group summaries with specific fields.

API Version
51.0

Requires Chatter
No

Signature
public static ConnectApi.OrderDeliveryGroupSummaryCollection
getOrderDeliveryGroupSummaries(String webstoreId, String effectiveAccountId, String orderSummaryId, List<String> fields, Integer pageSize, ConnectApi.OrderDeliveryGroupSummarySort sortParam)

Parameters
webstoreId
  Type: String
  ID of the webstore.

effectiveAccountId
  Type: String
  ID of the account for which the request is made. If null, defaults to the account ID for the context user.

orderSummaryId
  Type: String
  ID of the order summary.

fields
  Type: List<String>
  List of up to 15 order delivery group summary or order delivery method fields to display in the UI in each item row.

pageSize
  Type: Integer
  Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
  Type: ConnectApi.OrderDeliveryGroupSummarySort
  Sort order for order delivery group summaries. Values are:
  • IdAsc—Sorts by ID in ascending alphanumeric order (A–Z, 0–9).
  • IdDesc—Sorts by ID in descending alphanumeric order (Z–A, 9–0).
If `null`, `IdAsc` is the default sort order.

Return Value
Type: `ConnectApi.OrderDeliveryGroupSummaryCollection`

```java
getOrderItemSummaries(webstoreId, effectiveAccountId, orderSummaryId)
```
Get order item summaries.

API Version
51.0

Requires Chatter
No

Signature
```java
public static ConnectApi.OrderItemSummaryCollection getOrderItemSummaries(String webstoreId, String effectiveAccountId, String orderSummaryId)
```

Parameters
- `webstoreId`
  Type: `String`
  ID of the webstore.
- `effectiveAccountId`
  Type: `String`  
  ID of the account for which the request is made. If `null`, defaults to the account ID for the context user.
- `orderSummaryId`
  Type: `String`
  ID of the order summary.

Return Value
Type: `ConnectApi.OrderItemSummaryCollection`

```java
getOrderItemSummaries(webstoreId, effectiveAccountId, orderSummaryId, pageSize)
```
Get order item summaries.

API Version
51.0
Requires Chatter
No

Signature

```java
public static ConnectApi.OrderItemSummaryCollection getOrderItemSummaries(String webstoreId, String effectiveAccountId, String orderSummaryId, Integer pageSize)
```

Parameters

- **webstoreId**
  Type: `String`
  ID of the webstore.

- **effectiveAccountId**
  Type: `String`
  ID of the account for which the request is made. If `null`, defaults to the account ID for the context user.

- **orderSummaryId**
  Type: `String`
  ID of the order summary.

- **pageSize**
  Type: `Integer`
  Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

Return Value

Type: `ConnectApi.OrderItemSummaryCollection`

```java
getOrderItemSummaries(webstoreId, effectiveAccountId, orderSummaryId, orderDeliveryGroupSummaryId)
```

Get order item summaries for a delivery group summary.

API Version

51.0

Requires Chatter

No

Signature

```java
public static ConnectApi.OrderItemSummaryCollection getOrderItemSummaries(String webstoreId, String effectiveAccountId, String orderSummaryId, String orderDeliveryGroupSummaryId)
```
Parameters

webstoreId
  Type: String
  ID of the webstore.

effectiveAccountId
  Type: String
  ID of the account for which the request is made. If null, defaults to the account ID for the context user.

orderSummaryId
  Type: String
  ID of the order summary.

orderDeliveryGroupSummaryId
  Type: String
  ID of the order delivery group summary.

Return Value

Type: ConnectApi.OrderItemSummaryCollection

getOrderItemSummaries(webstoreId, effectiveAccountId, orderSummaryId, orderDeliveryGroupSummaryId, pageSize)

Get order item summaries for a delivery group summary.

API Version

S1.0

Requires Chatter

No

Signature

public static ConnectApi.OrderItemSummaryCollection getOrderItemSummaries(String webstoreId, String effectiveAccountId, String orderSummaryId, String orderDeliveryGroupSummaryId, Integer pageSize)

Parameters

webstoreId
  Type: String
  ID of the webstore.

effectiveAccountId
  Type: String
  ID of the account for which the request is made. If null, defaults to the account ID for the context user.
**orderSummaryId**  
Type: **String**  
ID of the order summary.

**orderDeliveryGroupSummaryId**  
Type: **String**  
ID of the order delivery group summary.

**pageSize**  
Type: **Integer**  
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in **null**, the default size is 25.

**Return Value**  
Type: **ConnectApi.OrderItemSummaryCollection**

getOrderItemSummaries(webstoreId, effectiveAccountId, orderSummaryId, orderDeliveryGroupSummaryId, pageParam)

Get a page of order item summaries for a delivery group summary.

**API Version**

$1.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.OrderItemSummaryCollection getOrderItemSummaries(String webstoreId, String effectiveAccountId, String orderSummaryId, String orderDeliveryGroupSummaryId, String pageParam)
```

**Parameters**

**webstoreId**  
Type: **String**  
ID of the webstore.

**effectiveAccountId**  
Type: **String**  
ID of the account for which the request is made. If **null**, defaults to the account ID for the context user.

**orderSummaryId**  
Type: **String**  
ID of the order summary.

**orderDeliveryGroupSummaryId**  
Type: **String**
ID of the order delivery group summary.

**pageParam**
Type: **String**
Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**Return Value**
Type: `ConnectApi.OrderItemSummaryCollection`

`getOrderItemSummaries(webstoreId, effectiveAccountId, orderSummaryId, orderDeliveryGroupSummaryId, fields)`
Get order item summaries for a delivery group summary with specific fields.

**API Version**
51.0

**Requires Chatter**
No

**Signature**

```java
public static ConnectApi.OrderItemSummaryCollection getOrderItemSummaries(String webstoreId, String effectiveAccountId, String orderSummaryId, String orderDeliveryGroupSummaryId, List<String> fields)
```

**Parameters**

- **webstoreId**
  Type: **String**
  ID of the webstore.

- **effectiveAccountId**
  Type: **String**
  ID of the account for which the request is made. If `null`, defaults to the account ID for the context user.

- **orderSummaryId**
  Type: **String**
  ID of the order summary.

- **orderDeliveryGroupSummaryId**
  Type: **String**
  ID of the order delivery group summary.

- **fields**
  Type: `List<String>`
  List of up to 15 order item summary or product fields to display in the UI in each item row.
Return Value
Type: `ConnectApi.OrderItemSummaryCollection`

```java
public static ConnectApi.OrderItemSummaryCollection getOrderItemSummaries(String webstoreId, String effectiveAccountId, String orderSummaryId, String orderDeliveryGroupSummaryId, List<String> fields, Integer pageSize)
```
Get order item summaries for a delivery group summary with specific fields.

**API Version**

51.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.OrderItemSummaryCollection getOrderItemSummaries(String webstoreId, String effectiveAccountId, String orderSummaryId, String orderDeliveryGroupSummaryId, List<String> fields, Integer pageSize)
```

**Parameters**

- `webstoreId`
  - Type: `String`
  - ID of the webstore.
- `effectiveAccountId`
  - Type: `String`
  - ID of the account for which the request is made. If `null`, defaults to the account ID for the context user.
- `orderSummaryId`
  - Type: `String`
  - ID of the order summary.
- `orderDeliveryGroupSummaryId`
  - Type: `String`
  - ID of the order delivery group summary.
- `fields`
  - Type: `List<String>`
  - List of up to 15 order item summary or product fields to display in the UI in each item row.
- `pageSize`
  - Type: `Integer`
  - Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**Return Value**

Type: `ConnectApi.OrderItemSummaryCollection`
getOrderItemSummaries(webstoreId, effectiveAccountId, orderSummaryId, orderDeliveryGroupSummaryId, fields, pageParam)

Get a page of order item summaries for a delivery group summary with specific fields.

API Version

$1.0

Requires Chatter

No

Signature

public static ConnectApi.OrderItemSummaryCollection getOrderItemSummaries(String webstoreId, String effectiveAccountId, String orderSummaryId, String orderDeliveryGroupSummaryId, List<String> fields, String pageParam)

Parameters

webstoreId
  Type: String
  ID of the webstore.

effectiveAccountId
  Type: String
  ID of the account for which the request is made. If null, defaults to the account ID for the context user.

orderSummaryId
  Type: String
  ID of the order summary.

orderDeliveryGroupSummaryId
  Type: String
  ID of the order delivery group summary.

fields
  Type: List<String>
  List of up to 15 order item summary or product fields to display in the UI in each item row.

pageParam
  Type: String
  Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in null, the first page is returned.

Return Value

Type: ConnectApi.OrderItemSummaryCollection
getOrderItemSummaries(webstoreId, effectiveAccountId, orderSummaryId, orderDeliveryGroupSummaryId, fields, sortParam)

Get a sorted list of order item summaries for a delivery group summary with specific fields.

API Version

$1.0

Requires Chatter

No

Signature

public static ConnectApi.OrderItemSummaryCollection getOrderItemSummaries(String webstoreId, String effectiveAccountId, String orderSummaryId, String orderDeliveryGroupSummaryId, List<String> fields, ConnectApi.OrderItemSummarySort sortParam)

Parameters

webstoreId
Type: String
ID of the webstore.

effectiveAccountId
Type: String
ID of the account for which the request is made. If null, defaults to the account ID for the context user.

orderSummaryId
Type: String
ID of the order summary.

orderDeliveryGroupSummaryId
Type: String
ID of the order delivery group summary.

fields
Type: List<String>
List of up to 15 order item summary or product fields to display in the UI in each item row.

sortParam
Type: ConnectApi.OrderItemSummarySort
Sort order for order item summaries. Values are:
- IdAsc—Sorts by ID in ascending alphanumeric order (A–Z, 0–9).
- IdDesc—Sorts by ID in descending alphanumeric order (Z–A, 9–0).
If null, IdAsc is the default sort order.
Return Value
Type: `ConnectApi.OrderItemSummaryCollection`

`getOrderItemSummaries(webstoreId, effectiveAccountId, orderSummaryId, orderDeliveryGroupSummaryId, fields, pageSize, sortParam)`
Get a sorted page of order item summaries for a delivery group summary with specific fields.

API Version
51.0

Requires Chatter
No

Signature
```
public static ConnectApi.OrderItemSummaryCollection getOrderItemSummaries(String webstoreId, String effectiveAccountId, String orderSummaryId, String orderDeliveryGroupSummaryId, List<String> fields, Integer pageSize, ConnectApi.OrderItemSummarySort sortParam)
```

Parameters
- **webstoreId**
  Type: `String`
  ID of the webstore.
- **effectiveAccountId**
  Type: `String`
  ID of the account for which the request is made. If `null`, defaults to the account ID for the context user.
- **orderSummaryId**
  Type: `String`
  ID of the order summary.
- **orderDeliveryGroupSummaryId**
  Type: `String`
  ID of the order delivery group summary.
- **fields**
  Type: `List<String>`
  List of up to 15 order item summary or product fields to display in the UI in each item row.
- **pageSize**
  Type: `Integer`
  Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.
- **sortParam**
  Type: `ConnectApi.OrderItemSummarySort`
Sort order for order item summaries. Values are:

- **IdAsc**—Sorts by ID in ascending alphanumeric order (A–Z, 0–9).
- **IdDesc**—Sorts by ID in descending alphanumeric order (Z–A, 9–0).

If **null**, **IdAsc** is the default sort order.

**Return Value**

Type: `ConnectApi.OrderItemSummaryCollection`

```java
getOrderItemSummaries(webstoreId, effectiveAccountId, orderSummaryId, orderDeliveryGroupSummaryId, fields, pageParam, pageSize, sortParam, includeAdjustmentDetails)
```

Get a sorted page of order item summaries for a delivery group summary with specific fields and include adjustment details.

**API Version**

56.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.OrderItemSummaryCollection getOrderItemSummaries(String webstoreId, String effectiveAccountId, String orderSummaryId, String orderDeliveryGroupSummaryId, List<String> fields, String pageParam, Integer pageSize, ConnectApi.OrderItemSummarySort sortParam, Boolean includeAdjustmentDetails)
```

**Parameters**

- **webstoreId**
  Type: `String`
  ID of the webstore.
- **effectiveAccountId**
  Type: `String`
  ID of the account for which the request is made. If **null**, defaults to the account ID for the context user.
- **orderSummaryId**
  Type: `String`
  ID of the order summary.
- **orderDeliveryGroupSummaryId**
  Type: `String`
  ID of the order delivery group summary.
- **fields**
  Type: `List<String>`
List of up to 15 order item summary or product fields to display in the UI in each item row.

**pageParam**
Type: `String`
Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**
Type: `Integer`
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**sortParam**
Type: `ConnectApi.OrderItemSummarySort`
Sort order for order item summaries. Values are:
- `IdAsc`—Sorts by ID in ascending alphanumeric order (A–Z, 0–9).
- `IdDesc`—Sorts by ID in descending alphanumeric order (Z–A, 9–0).
If `null`, `IdAsc` is the default sort order.

**includeAdjustmentDetails**
Type: `Boolean`
Specifies whether to return adjustment details (true) or not (false).

### Return Value
Type: `ConnectApi.OrderItemSummaryCollection`

#### getOrderItemSummaryAdjustments (webstoreId, orderSummaryId, orderItemSummaryAdjustmentCollectionInput)
Get adjustments for order items.

### API Version
53.0

### Requires Chatter
No

### Signature
```
public static ConnectApi.OrderItemSummaryAdjustmentCollection
getOrderItemSummaryAdjustments(String webstoreId, String orderSummaryId,
ConnectApi.OrderItemSummaryAdjustmentCollectionInput
orderItemSummaryAdjustmentCollectionInput)
```

### Parameters
- **webstoreId**
  Type: `String`
ID of the webstore.

`orderSummaryId`  
Type: `String`  
ID of the order summary.

`orderItemSummaryAdjustmentCollectionInput`  
Type: `ConnectApi.OrderItemSummaryAdjustmentCollectionInput`  
Collection of order item summaries to get adjustments for.

**Return Value**

Type: `ConnectApi.OrderItemSummaryAdjustmentCollection`

```java
getOrderItemSummaryAdjustments(String webstoreId, String orderSummaryId, 
ConnectApi.OrderItemSummaryAdjustmentCollectionInput, String effectiveAccountId)
```

Get adjustments for order items.

**API Version**

53.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.OrderItemSummaryAdjustmentCollection 
getOrderItemSummaryAdjustments(String webstoreId, String orderSummaryId, 
ConnectApi.OrderItemSummaryAdjustmentCollectionInput orderItemSummaryAdjustmentCollectionInput, String effectiveAccountId)
```

**Parameters**

`webstoreId`  
Type: `String`  
ID of the webstore.

`orderSummaryId`  
Type: `String`  
ID of the order summary.

`orderItemSummaryAdjustmentCollectionInput`  
Type: `ConnectApi.OrderItemSummaryAdjustmentCollectionInput`  
Collection of order item summaries to get adjustments for.

`effectiveAccountId`  
Type: `String`  
ID of the account for which the request is made. If `null`, defaults to the account ID for the context user.
Return Value
Type: `ConnectApi.OrderItemSummaryAdjustmentCollection`

`getOrderShipmentItems(webstoreId, shipmentId)`
Get order shipment items.

API Version
52.0

Requires Chatter
No

Signature
```java
public static ConnectApi.OrderShipmentItemCollection getOrderShipmentItems(String webstoreId, String shipmentId)
```

Parameters
- `webstoreId`
  Type: `String`
  ID of the webstore.
- `shipmentId`
  Type: `String`
  ID of the shipment.

Return Value
Type: `ConnectApi.OrderShipmentItemCollection`

`getOrderShipmentItems(webstoreId, shipmentId, effectiveAccountId)`
Get order shipment items.

API Version
52.0

Requires Chatter
No

Signature
```java
public static ConnectApi.OrderShipmentItemCollection getOrderShipmentItems(String webstoreId, String shipmentId, String effectiveAccountId)
```
Parameters

`webstoreId`
Type: `String`
ID of the webstore.

`shipmentId`
Type: `String`
ID of the shipment.

`effectiveAccountId`
Type: `String`
ID of the account for which the request is made. If `null`, defaults to the account ID for the context user.

Return Value
Type: `ConnectApi.OrderShipmentItemCollection`

`getOrderShipmentItems(webstoreId, shipmentId, effectiveAccountId, fields)`
Get order shipment items with specific fields.

API Version
52.0

Requires Chatter
No

Signature

```java
public static ConnectApi.OrderShipmentItemCollection getOrderShipmentItems(String webstoreId, String shipmentId, String effectiveAccountId, List<String> fields)
```

Parameters

`webstoreId`
Type: `String`
ID of the webstore.

`shipmentId`
Type: `String`
ID of the shipment.

`effectiveAccountId`
Type: `String`
ID of the account for which the request is made. If `null`, defaults to the account ID for the context user.

`fields`
Type: `List<String>`
List of up to 15 additional shipment items, order item summary, and product fields to display in the UI in each item row.
Return Value
Type: ConnectApi.OrderShipmentItemCollection

getOrderShipmentItems(webstoreId, shipmentId, effectiveAccountId, fields, pageToken, pageSize)
Get a page of order shipment items with specific fields.

API Version
52.0

Requires Chatter
No

Signature
public static ConnectApi.OrderShipmentItemCollection getOrderShipmentItems(String webstoreId, String shipmentId, String effectiveAccountId, List<String> fields, String pageToken, Integer pageSize)

Parameters
webstoreId
Type: String
ID of the webstore.

shipmentId
Type: String
ID of the shipment.

effectiveAccountId
Type: String
ID of the account for which the request is made. If null, defaults to the account ID for the context user.

fields
Type: List<String>
List of up to 15 additional shipment items, order item summary, and product fields to display in the UI in each item row.

pageToken
Type: String
Specifies the base64 encoded page token. Page tokens are returned as part of the response. If unspecified, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

Return Value
Type: ConnectApi.OrderShipmentItemCollection
**getOrderShipmentItems(webstoreId, shipmentId, effectiveAccountId, fields, pageToken, pageSize, sortOrder)**

Get a sorted page of order shipment items.

**API Version**

52.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.OrderShipmentItemCollection getOrderShipmentItems(String webstoreId, String shipmentId, String effectiveAccountId, List<String> fields, String pageToken, Integer pageSize, ConnectApi.OrderShipmentItemSort sortOrder)
```

**Parameters**

- **webstoreId**
  
  Type: `String`
  
  ID of the webstore.

- **shipmentId**
  
  Type: `String`
  
  ID of the shipment.

- **effectiveAccountId**
  
  Type: `String`
  
  ID of the account for which the request is made. If `null`, defaults to the account ID for the context user.

- **fields**
  
  Type: `List<String>`
  
  List of up to 15 additional shipment items, order item summary, and product fields to display in the UI in each item row.

- **pageToken**
  
  Type: `String`
  
  Specifies the base64 encoded page token. Page tokens are returned as part of the response. If unspecified, the first page is returned.

- **pageSize**
  
  Type: `Integer`
  
  Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

- **sortOrder**
  
  Type: `ConnectApi.OrderShipmentItemSort`
  
  Sort order for order shipment items. Values are:
  
  - `IdAsc`—Sorts by ID in ascending alphanumeric order (A–Z, 0–9).
  - `IdDesc`—Sorts by ID in descending alphanumeric order (Z–A, 9–0).
  
  If unspecified, defaults to `IdAsc`. 

Return Value
Type: `ConnectApi.OrderShipmentItemCollection`

`getOrderShipments(webstoreId, orderSummaryId)`
Get order shipments.

API Version
52.0

Requires Chatter
No

Signature
```java
public static ConnectApi.OrderShipmentCollection getOrderShipments(String webstoreId,
String orderSummaryId)
```

Parameters
- `webstoreId`
  Type: `String`
  ID of the webstore.
- `orderSummaryId`
  Type: `String`
  ID of the order summary.

Return Value
Type: `ConnectApi.OrderShipmentCollection`

`getOrderShipments(webstoreId, orderSummaryId, effectiveAccountId)`
Get order shipments.

API Version
52.0

Requires Chatter
No

Signature
```java
public static ConnectApi.OrderShipmentCollection getOrderShipments(String webstoreId,
String orderSummaryId, String effectiveAccountId)
```
Parameters

webstoreId
Type: String
ID of the webstore.

orderSummaryId
Type: String
ID of the order summary.

effectiveAccountId
Type: String
ID of the account for which the request is made. If null, defaults to the account ID for the context user.

Return Value

Type: ConnectApi.OrderShipmentCollection

getOrderShipments(webstoreId, orderSummaryId, effectiveAccountId, fields)
Get order shipments with specific fields.

API Version

52.0

Requires Chatter

No

Signature

public static ConnectApi.OrderShipmentCollection getOrderShipments(String webstoreId, String orderSummaryId, String effectiveAccountId, List<String> fields)

Parameters

webstoreId
Type: String
ID of the webstore.

orderSummaryId
Type: String
ID of the order summary.

effectiveAccountId
Type: String
ID of the account for which the request is made. If null, defaults to the account ID for the context user.

fields
Type: List<String>
List of up to 15 additional shipment and order delivery method fields to display in the UI in each item row.
Return Value
Type: `ConnectApi.OrderShipmentCollection`

```java
public static ConnectApi.OrderShipmentCollection getOrderShipments(String webstoreId, String orderSummaryId, String effectiveAccountId, List<String> fields, Integer pageSize, String pageToken)
```

Get a page of order shipments with specific fields.

API Version
52.0

Requires Chatter
No

Signature

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>webstoreId</code></td>
<td><code>String</code></td>
<td>ID of the webstore.</td>
</tr>
<tr>
<td><code>orderSummaryId</code></td>
<td><code>String</code></td>
<td>ID of the order summary.</td>
</tr>
<tr>
<td><code>effectiveAccountId</code></td>
<td><code>String</code></td>
<td>ID of the account for which the request is made. If <code>null</code>, defaults to the account ID for the context user.</td>
</tr>
<tr>
<td><code>fields</code></td>
<td><code>List&lt;String&gt;</code></td>
<td>List of up to 15 additional shipment and order delivery method fields to display in the UI in each item row.</td>
</tr>
<tr>
<td><code>pageSize</code></td>
<td><code>Integer</code></td>
<td>Specifies the number of items per page. Valid values are from 1 through 100. If you pass in <code>null</code>, the default size is 25.</td>
</tr>
<tr>
<td><code>pageToken</code></td>
<td><code>String</code></td>
<td>Specifies the base64 encoded page token. Page tokens are returned as part of the response. If unspecified, the first page is returned.</td>
</tr>
</tbody>
</table>

Return Value
Type: `ConnectApi.OrderShipmentCollection`
getOrderShipments(webstoreId, orderSummaryId, effectiveAccountId, fields, pageSize, pageToken, sortOrder)

Get a sorted page of order shipments with specific fields.

API Version

52.0

Requires Chatter

No

Signature

public static ConnectApi.OrderShipmentCollection getOrderShipments(String webstoreId, String orderSummaryId, String effectiveAccountId, List<String> fields, Integer pageSize, String pageToken, ConnectApi.OrderShipmentSort sortOrder)

Parameters

webstoreId
Type: String
ID of the webstore.

orderSummaryId
Type: String
ID of the order summary.

effectiveAccountId
Type: String
ID of the account for which the request is made. If null, defaults to the account ID for the context user.

fields
Type: List<String>
List of up to 15 additional shipment and order delivery method fields to display in the UI in each item row.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

pageToken
Type: String
Specifies the base64 encoded page token. Page tokens are returned as part of the response. If unspecified, the first page is returned.

sortOrder
Type: ConnectApi.OrderShipmentSort
Sort order for order shipments. Values are:

- ExpectedDeliveryDateAsc—Sorts by the oldest expected delivery date.
- ExpectedDeliveryDateDesc—Sorts by the most recent expected delivery date.
- ShipmentNumberAsc—Sorts by shipment number in ascending order (0–9).
• ShipmentNumberDesc—Sorts by shipment number in descending order (9–0).
If unspecified, defaults to ShipmentNumberAsc.
If you’re sorting by expected delivery date, make sure the expected delivery date is populated on your shipment records. A null value isn’t supported and results in an error.

Return Value
Type: ConnectApi.OrderShipmentCollection

getOrderSummaries(webstoreId)
Get order summaries.

API Version
51.0

Requires Chatter
No

Signature
public static ConnectApi.OrderSummaryCollectionRepresentation getOrderSummaries(String webstoreId)

Parameters
webstoreId
  Type: String
  ID of the webstore.

Return Value
Type: ConnectApi.OrderSummaryCollectionRepresentation

getOrderSummaries(webstoreId, effectiveAccountId)
Get order summaries.

API Version
51.0

Requires Chatter
No
public static ConnectApi.OrderSummaryCollectionRepresentation getOrderSummaries(String webstoreId, String effectiveAccountID)

Parameters

webstoreId
Type: String
ID of the webstore.
effectiveAccountID
Type: String
ID of the account for which the request is made. If null, defaults to the account ID for the context user.

Return Value

Type: ConnectApi.OrderSummaryCollectionRepresentation

getOrderSummaries(webstoreId, effectiveAccountID, fields)
Get order summaries with specific fields.

API Version

51.0

Requires Chatter

No

Signature

public static ConnectApi.OrderSummaryCollectionRepresentation getOrderSummaries(String webstoreId, String effectiveAccountID, List<String> fields)

Parameters

webstoreId
Type: String
ID of the webstore.
effectiveAccountID
Type: String
ID of the account for which the request is made. If null, defaults to the account ID for the context user.

fields
Type: List<String>
List of up to 35 additional order summary fields to display in the UI in each item row.
These order summary fields are returned regardless of fields specified.
getOrderSummaries(webstoreId, effectiveAccountId, fields, pageSize, pageToken)

Get a page of order summaries with specific fields.

API Version
51.0

Requires Chatter
No

Signature
public static ConnectApi.OrderSummaryCollectionRepresentation getOrderSummaries(String webstoreId, String effectiveAccountId, List<String> fields, Integer pageSize, String pageToken)

Parameters

webstoreId
Type: String
ID of the webstore.

effectiveAccountId
Type: String
ID of the account for which the request is made. If null, defaults to the account ID for the context user.

fields
Type: List<String>
List of up to 35 additional order summary fields to display in the UI in each item row.

These order summary fields are returned regardless of fields specified.

- createdDate
- orderSummaryId
- orderNumber
• orderedDate
• ownerID
• status
• totalAmount

**pageSize**
Type: **Integer**
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**pageToken**
Type: **String**
Specifies the base64 encoded page token. Page tokens are returned as part of the response. If unspecified, the first page is returned.

**Return Value**
Type: `ConnectApi.OrderSummaryCollectionRepresentation`

**getOrderSummaries(webstoreId, effectiveAccountId, fields, pageSize, pageToken, sortOrder)**
Get a sorted page of order summaries with specific fields.

**API Version**
**51.0**

**Requires Chatter**
No

**Signature**
`public static ConnectApi.OrderSummaryCollectionRepresentation getOrderSummaries(String webstoreId, String effectiveAccountId, List<String> fields, Integer pageSize, String pageToken, ConnectApi.OrderSummarySortOrder sortOrder)`

**Parameters**

**webstoreId**
Type: **String**
ID of the webstore.

**effectiveAccountId**
Type: **String**
ID of the account for which the request is made. If `null`, defaults to the account ID for the context user.

**fields**
Type: `List<String>`
List of up to 35 additional order summary fields to display in the UI in each item row.
These order summary fields are returned regardless of fields specified.
• createdDate
• orderSummaryId
• orderNumber
• orderedDate
• ownerId
• status
• totalAmount

**pageSize**
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**pageToken**
Type: String
Specifies the base64 encoded page token. Page tokens are returned as part of the response. If unspecified, the first page is returned.

**sortOrder**
Type: `ConnectApi.OrderSummarySortOrder`
Sort order for order summaries. Values are:
• CreatedDateAsc—Sorts by the oldest created date.
• CreatedDateDesc—Sorts by the most recent created date.
• OrderedDateAsc—Sorts by the oldest ordered date.
• OrderedDateDesc—Sorts by the most recent ordered date.
If unspecified, defaults to `OrderedDateDesc`.
If you’re sorting by ordered date, make sure the ordered date is populated on your order summary records. A `null` value isn’t supported and results in an error.

**Return Value**
Type: `ConnectApi.OrderSummaryCollectionRepresentation`

`getOrderSummaries(webstoreId, effectiveAccountId, fields, pageSize, pageToken, sortOrder, earliestDate, latestDate)`
Get a sorted page of order summaries with specific fields within a specific date range.

**API Version**
51.0

**Requires Chatter**
No
Signature

public static ConnectApi.OrderSummaryCollectionRepresentation getOrderSummaries(String webstoreId, String effectiveAccountId, List<String> fields, Integer pageSize, String pageToken, ConnectApi.OrderSummarySortOrder sortOrder, String earliestDate, String latestDate)

Parameters

webstoreId
Type: String
ID of the webstore.

effectiveAccountId
Type: String
ID of the account for which the request is made. If null, defaults to the account ID for the context user.

fields
Type: List<String>
List of up to 35 additional order summary fields to display in the UI in each item row. These order summary fields are returned regardless of fields specified.

• createdDate
• orderSummaryId
• orderNumber
• orderedDate
• ownerId
• status
• totalAmount

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

pageToken
Type: String
Specifies the base64 encoded page token. Page tokens are returned as part of the response. If unspecified, the first page is returned.

sortOrder
Type: ConnectApi.OrderSummarySortOrder
Sort order for order summaries. Values are:

• CreatedDateAsc—Sorts by the oldest created date.
• CreatedDateDesc—Sorts by the most recent created date.
• OrderedDateAsc—Sorts by the oldest ordered date.
• OrderedDateDesc—Sorts by the most recent ordered date.

If unspecified, defaults to OrderedDateDesc.

If you’re sorting by ordered date, make sure the ordered date is populated on your order summary records. A null value isn’t supported and results in an error.
earliestDate
Type: String
Oldest created or ordered date, depending on the sortOrder value, for order summaries to return. Results include any orders on and after this date. Expected format is an ISO 8601 date string, for example, 2020-02-25T18:24:31.000Z.

latestDate
Type: String
Most recent created or ordered date, depending on the sortOrder value, for order summaries to return. Results include any orders before this date. Expected format is an ISO 8601 date string, for example, 2020-02-25T18:24:31.000Z.

Return Value
Type: ConnectApi.OrderSummaryCollectionRepresentation

getOrderSummaries(webstoreId, effectiveAccountId, fields, pageSize, pageToken, sortOrder, earliestDate, latestDate, ownerScoped)
Get a sorted page of order summaries with specific fields within a specific date range and scoped to orders owned by the context user.

API Version
S1.0

Requires Chatter
No

Signature
public static ConnectApi.OrderSummaryCollectionRepresentation getOrderSummaries(String webstoreId, String effectiveAccountId, List<String> fields, Integer pageSize, String pageToken, ConnectApi.OrderSummarySortOrder sortOrder, String earliestDate, String latestDate, Boolean ownerScoped)

Parameters
webstoreId
Type: String
ID of the webstore.

effectiveAccountId
Type: String
ID of the account for which the request is made. If null, defaults to the account ID for the context user.

fields
Type: List<String>
List of up to 35 additional order summary fields to display in the UI in each item row.
These order summary fields are returned regardless of fields specified.
• createdDate
- orderSummaryId
- orderNumber
- orderedDate
- ownerId
- status
- totalAmount

**pageSize**
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

**pageToken**
Type: String
Specifies the base64 encoded page token. Page tokens are returned as part of the response. If unspecified, the first page is returned.

**sortOrder**
Type: ConnectApi.OrderSummarySortOrder
Sort order for order summaries. Values are:
- CreatedDateAsc—Sorts by the oldest created date.
- CreatedDateDesc—Sorts by the most recent created date.
- OrderedDateAsc—Sorts by the oldest ordered date.
- OrderedDateDesc—Sorts by the most recent ordered date.
If unspecified, defaults to OrderedDateDesc.
If you’re sorting by ordered date, make sure the ordered date is populated on your order summary records. A null value isn’t supported and results in an error.

**earliestDate**
Type: String
Oldest created or ordered date, depending on the sortOrder value, for order summaries to return. Results include any orders on and after this date. Expected format is an ISO 8601 date string, for example, 2020-02-25T18:24:31.000Z.

**latestDate**
Type: String
Most recent created or ordered date, depending on the sortOrder value, for order summaries to return. Results include any orders before this date. Expected format is an ISO 8601 date string, for example, 2020-02-25T18:24:31.000Z.

**ownerScoped**
Type: Boolean
Specifies whether the results are scoped to orders owned by the context user (true) or to orders owned by and shared with the context user (false). If unspecified, defaults to true.

**Return Value**
Type: ConnectApi.OrderSummaryCollectionRepresentation
getOrderSummaries(webstoreId, effectiveAccountId, fields, pageSize, pageToken, sortOrder, earliestDate, latestDate, ownerScoped, includeAdjustmentDetails)

Get a sorted page of order summaries with specific fields within a specific date range and scoped to orders owned by the context user.

API Version

S1.0

Requires Chatter

No

Signature

public static ConnectApi.OrderSummaryCollectionRepresentation getOrderSummaries(String webstoreId, String effectiveAccountId, List<String> fields, Integer pageSize, String pageToken, ConnectApi.OrderSummarySortOrder sortOrder, String earliestDate, String latestDate, Boolean ownerScoped, Boolean includeAdjustmentDetails)

Parameters

webstoreId
  Type: String
  ID of the webstore.

effectiveAccountId
  Type: String
  ID of the account for which the request is made. If null, defaults to the account ID for the context user.

fields
  Type: List<String>
  List of up to 35 additional order summary fields to display in the UI in each item row.
  These order summary fields are returned regardless of fields specified.
  • createdDate
  • orderSummaryId
  • orderNumber
  • orderedDate
  • ownerId
  • status
  • totalAmount

pageSize
  Type: Integer
  Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

pageToken
  Type: String
  Specifies the base64 encoded page token. Page tokens are returned as part of the response. If unspecified, the first page is returned.
sortOrder
Type: `ConnectApi.OrderSummarySortOrder`
Sort order for order summaries. Values are:
- CreatedDateAsc—Sorts by the oldest created date.
- CreatedDateDesc—Sorts by the most recent created date.
- OrderedDateAsc—Sorts by the oldest ordered date.
- OrderedDateDesc—Sorts by the most recent ordered date.
If unspecified, defaults to OrderedDateDesc.
If you’re sorting by ordered date, make sure the ordered date is populated on your order summary records. A null value isn’t supported and results in an error.

earliestDate
Type: String
Oldest created or ordered date, depending on the sortOrder value, for order summaries to return. Results include any orders on and after this date. Expected format is an ISO 8601 date string, for example, 2020-02-25T18:24:31.000Z.

latestDate
Type: String
Most recent created or ordered date, depending on the sortOrder value, for order summaries to return. Results include any orders before this date. Expected format is an ISO 8601 date string, for example, 2020-02-25T18:24:31.000Z.

ownerScoped
Type: Boolean
Specifies whether the results are scoped to orders owned by the context user (`true`) or to orders owned by and shared with the context user (`false`). If unspecified, defaults to `true`.

includeAdjustmentDetails
Type: Boolean
Specifies whether to fetch price adjustment details based on their type (`true`). If unspecified, defaults to `false`.

Return Value
Type: `ConnectApi.OrderSummaryCollectionRepresentation`

`getOrderSummary(webstoreId, orderSummaryId, effectiveAccountId)`
Get an order summary.

API Version
5.5.0

Requires Chatter
No
public static ConnectApi.OrderSummaryRepresentation getOrderSummary(String webstoreId, String orderSummaryId, String effectiveAccountId)

Parameters

webstoreId
Type: String
ID of the webstore.

orderSummaryId
Type: String
ID of the order summary.

effectiveAccountId
Type: String
ID of the account for which the request is made. If null, defaults to the account ID for the context user.

Return Value
Type: ConnectApi.OrderSummaryRepresentation

getOrderSummary(webstoreId, orderSummaryId, effectiveAccountId, fields)
Get an order summary with fields.

API Version
55.0

Requires Chatter
No

Signature
public static ConnectApi.OrderSummaryRepresentation getOrderSummary(String webstoreId, String orderSummaryId, String effectiveAccountId, List<String> fields)

Parameters

webstoreId
Type: String
ID of the webstore.

orderSummaryId
Type: String
ID of the order summary.

effectiveAccountId
Type: String
ID of the account for which the request is made. If null, defaults to the account ID for the context user.

**fields**

Type: `List<String>`

List of up to 35 additional order summary fields to display in the UI in each item row.

These order summary fields are returned regardless of fields specified.

- createdDate
- orderSummaryId
- orderNumber
- orderedDate
- ownerId
- status
- totalAmount

**Return Value**

Type: `ConnectApi.OrderSummaryRepresentation`

**getOrderSummary(webstoreId, orderSummaryId, effectiveAccountId, fields, includeAdjustmentDetails)**

Get an order summary with fields and include adjustment details.

**API Version**

55.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.OrderSummaryRepresentation getOrderSummary(String webstoreId, String orderSummaryId, String effectiveAccountId, List<String> fields, Boolean includeAdjustmentDetails)
```

**Parameters**

- `webstoreId`
  
  Type: `String`
  
  ID of the webstore.

- `orderSummaryId`
  
  Type: `String`
  
  ID of the order summary.

- `effectiveAccountId`
  
  Type: `String`
ID of the account for which the request is made. If **null**, defaults to the account ID for the context user.

**fields**

Type: **List<String>**

List of up to 35 additional order summary fields to display in the UI in each item row.

These order summary fields are returned regardless of fields specified.

- createdDate
- orderSummaryId
- orderNumber
- orderedDate
- ownerId
- status
- totalAmount

**includeAdjustmentDetails**

Type: **Boolean**

Specifies whether to return adjustment details (**true**) or not (**false**). If unspecified, the default value is **false**.

**Return Value**

Type: **ConnectApi.OrderSummaryRepresentation**

`getOrderSummaryAdjustments(webstoreId, orderSummaryId)`

Get adjustments for an order summary.

**API Version**

53.0

**Requires Chatter**

No

**Signature**

`public static ConnectApi.OrderSummaryAdjustmentCollection getOrderSummaryAdjustments(String webstoreId, String orderSummaryId)`

**Parameters**

- **webstoreId**
  Type: **String**
  ID of the webstore.

- **orderSummaryId**
  Type: **String**
  ID of the order summary.
Return Value
Type: ConnectApi.OrderSummaryAdjustmentCollection

getOrderSummaryAdjustments(webstoreId, orderSummaryId, effectiveAccountId)
Get adjustments for an order summary.

API Version
53.0

Requires Chatter
No

Signature
public static ConnectApi.OrderSummaryAdjustmentCollection
getOrderSummaryAdjustments(String webstoreId, String orderSummaryId, String
effectiveAccountId)

Parameters
webstoreId
Type: String
ID of the webstore.

orderSummaryId
Type: String
ID of the order summary.

effectiveAccountId
Type: String
ID of the account for which the request is made. If null, defaults to the account ID for the context user.

Return Value
Type: ConnectApi.OrderSummaryAdjustmentCollection

lookupOrderSummary(webstoreId, effectiveAccountId, fields, excludeLineItems,
excludeDeliveryGroups, excludeAdjustmentAggregates, excludeAdjustments,
deliveryGroupId, orderSummaryLookupInput) (Developer Preview)
Look up details about an order summary for a guest shopper or a registered buyer using the effective account ID, requested fields, line items, delivery groups, adjustments aggregates, and adjustments.

Note: This API is available as a developer preview. It isn’t generally available unless or until Salesforce announces its general availability in documentation or in press releases or public statements. All commands, parameters, and other features are subject to change or deprecation at any time, with or without notice. Don’t implement functionality developed with these commands or tools.
API Version
58.0

Available to Guest Users
58.0

Requires Chatter
No

Signature
public static ConnectApi.OrderSummaryLookupOutput lookupOrderSummary(String webstoreId, 
String effectiveAccountId, List<String> fields, Boolean excludeLineItems, Boolean 
excludeDeliveryGroups, Boolean excludeAdjustmentAggregates, Boolean excludeAdjustments, 
String deliveryGroupId, ConnectApi.OrderSummaryLookupInput orderSummaryLookupInput)

Parameters
webstoreId
  Type: String
  ID of the webstore.

effectiveAccountId
  Type: String
  ID of the account for which the request is made. If unspecified, defaults to the account ID for the context user or, for guest users, the 
guest buyer profile ID of the current store.

fields
  Type: List<String>
  List of specific fields, including custom fields, to return in the response along with default fields. For example, 
  OrderSummary.TotalAmount, OrderItemSummary.Quantity, Product2.Description, 

excludeLineItems
  Type: Boolean
  Specifies whether to exclude line items from the response. If unspecified, the default value is false.

excludeDeliveryGroups
  Type: Boolean
  Specifies whether to exclude delivery groups from the response. If unspecified, the default value is false.

excludeAdjustmentAggregates
  Type: Boolean
  Specifies whether to exclude adjustment aggregates associated with an order summary. Adjustment aggregates include fields 
detailing promotional amounts by price, tax, and total. Aggregates are calculated asynchronously and results returned to the order 
summary. If unspecified, the default value is false.

excludeAdjustments
  Type: Boolean
Specifies whether to exclude adjustments associated with an order summary. Adjustments include promotional discounts. If unspecified, the default value is false.

deliveryGroupId
Type: String
ID of the delivery group associated with the order summary.

orderSummaryLookupInput
Type: ConnectApi.OrderSummaryLookupInput
Order summary lookup input representation.

Return Value
Type: ConnectApi.OrderSummaryLookupOutput

lookupOrderSummary(webstoreId, effectiveAccountId, fields, orderSummaryLookupInput) (Developer Preview)
Look up details about an order summary for a guest shopper or a registered buyer using the effective account ID and requested fields.

Note: This API is available as a developer preview. It isn’t generally available unless or until Salesforce announces its general availability in documentation or in press releases or public statements. All commands, parameters, and other features are subject to change or deprecation at any time, with or without notice. Don’t implement functionality developed with these commands or tools.

API Version
58.0

Available to Guest Users
58.0

Requires Chatter
No

Signature
public static ConnectApi.OrderSummaryLookupOutput lookupOrderSummary(String webstoreId, String effectiveAccountId, List<String> fields, ConnectApi.OrderSummaryLookupInput orderSummaryLookupInput)

Parameters
webstoreId
Type: String
ID of the webstore.
effectiveAccountId
Type: String
ID of the account for which the request is made. If unspecified, defaults to the account ID for the context user or, for guest users, the guest buyer profile ID of the current store.

**fields**
Type: `List<String>`
List of specific fields, including custom fields, to return in the response along with default fields. For example, `OrderSummary.TotalAmount`, `OrderItemSummary.Quantity`, `Product2.Description`, `OrderDeliveryGroupSummary.GrandTotalAmount`, `OrderDeliveryMethod.Carrier`.

**orderSummaryLookupInput**
Type: `ConnectApi.OrderSummaryLookupInput`
Order summary lookup input representation.

**Return Value**
Type: `ConnectApi.OrderSummaryLookupOutput`

`lookupOrderSummary(webstoreId, effectiveAccountId, orderSummaryLookupInput)` (Developer Preview)
Look up details about an order summary for a guest shopper or a registered buyer using the effective account ID.

⚠️ **Note:** This API is available as a developer preview. It isn’t generally available unless or until Salesforce announces its general availability in documentation or in press releases or public statements. All commands, parameters, and other features are subject to change or deprecation at any time, with or without notice. Don’t implement functionality developed with these commands or tools.

**API Version**
58.0

**Available to Guest Users**
58.0

**Requires Chatter**
No

**Signature**
```
public static ConnectApi.OrderSummaryLookupOutput lookupOrderSummary(String webstoreId, String effectiveAccountId, ConnectApi.OrderSummaryLookupInput orderSummaryLookupInput)
```

**Parameters**
- `webstoreId`
  Type: `String`
  ID of the webstore.
effectiveAccountId
- Type: String
  ID of the account for which the request is made. If unspecified, defaults to the account ID for the context user or, for guest users, the guest buyer profile ID of the current store.

orderSummaryLookupInput
- Type: ConnectApi.OrderSummaryLookupInput
  Order summary lookup input representation.

Return Value
- Type: ConnectApi.OrderSummaryLookupOutput

updateCommerceAccountAddress(webstoreId, accountId, addressId, addressInput)
Update a Commerce account address for a webstore.

API Version
54.0

Requires Chatter
No

Signature
public static ConnectApi.CommerceAddressOutput updateCommerceAccountAddress(String webstoreId, String accountId, String addressId, ConnectApi.commerceAddressInput addressInput)

Parameters
- webstoreId
  - Type: String
    ID of the webstore.
- accountId
  - Type: String
    ID of the account.
- addressId
  - Type: String
    ID of the address.
- addressInput
  - Type: ConnectApi.commerceAddressInput
    Information about the address fields you want to update.
Return Value
Type: `ConnectApi.CommerceAddressOutput`

**CommerceCart Class**

Get, create, update, and delete carts. Get cart items, add items to carts, update and delete cart items.

**Namespace**

`ConnectApi`

**CommerceCart Methods**

These methods are for `CommerceCart`. All methods are static.

**IN THIS SECTION:**

- `addItemToCart(webstoreId, effectiveAccountId, activeCartOrId, cartItemInput, currencyIsoCode)`
  Add an item to a cart of a specific currency.

- `addItemToCart(webstoreId, effectiveAccountId, activeCartOrId, cartItems)`
  Add a batch of up to 100 items to a cart.

- `addItemToCart(webstoreId, effectiveAccountId, activeCartOrId, cartItems, currencyIsoCode)`
  Add a batch of up to 100 items to a cart of a specific currency.

- `addItemToCart(webstoreId, effectiveAccountId, activeCartOrId, cartItemInput)`
  Add an item to a cart.

- `applyCartCoupon(webstoreId, effectiveAccountId, activeCartOrId, cartCouponInput)`
  Apply a coupon to a cart.

- `applyCartCoupon(webstoreId, effectiveAccountId, activeCartOrId, cartCouponInput, currencyIsoCode)`
  Apply a coupon to a cart.

- `copyCartToWishlist(webstoreId, effectiveAccountId, activeCartOrId, cartToWishlistInput)`
  Copy the products from a cart to a wishlist.

- `createCart(webstoreId, cart)`
  Create a cart.

- `deleteCart(webstoreId, effectiveAccountId, activeCartOrId)`
  Delete a cart.

- `deleteCartCoupon(webstoreId, effectiveAccountId, activeCartOrId, cartCouponId)`
  Delete a coupon from a cart.

- `deleteCartCoupon(webstoreId, effectiveAccountId, activeCartOrId, cartCouponId, currencyIsoCode)`
  Delete a coupon from a cart.

- `deleteCartItem(webstoreId, effectiveAccountId, activeCartOrId, cartItemId)`
  Delete an item from a cart.

- `deleteInventoryReservation(webstoreId, activeCartOrId, effectiveAccountId) (Pilot)`
  Delete an inventory reservation.
getCartCoupons(webstoreId, effectiveAccountId, activeCartOrId)
Get coupons for a cart.
getCartCoupons(webstoreId, effectiveAccountId, activeCartOrId, currencyIsoCode)
Get coupons for a cart.
getCartItemPromotion(webstoreId, effectiveAccountId, activeCartOrId, cartItemPromotionCollectionInput)
Get promotions for a cart item.
getCartItemPromotion(webstoreId, effectiveAccountId, activeCartOrId, cartItemPromotionCollectionInput, currencyIsoCode)
Get a promotion for a cart item.
getCartItems(webstoreId, effectiveAccountId, activeCartOrId)
Get items in a cart.
getCartItems(webstoreId, effectiveAccountId, activeCartOrId, pageParam)
Get a page of items in a cart.
getCartItems(webstoreId, effectiveAccountId, activeCartOrId, pageParam, sortParam)
Get a sorted page of items in a cart.
getCartItems(webstoreId, effectiveAccountId, activeCartOrId, pageParam, pageSize)
Get a specified size page of items in a cart.
getCartItems(webstoreId, effectiveAccountId, activeCartOrId, pageParam, pageSize, sortParam)
Get a specified size, sorted page of items in a cart.
getCartItems(webstoreId, effectiveAccountId, activeCartOrId, pageParam, pageSize, sortParam, currencyIsoCode)
Get a specified size, sorted page of items filtered by product fields in a cart.
getCartItems(webstoreId, effectiveAccountId, activeCartOrId, pageParam, pageSize, sortParam, currencyIsoCode, includePromotions, includeCoupons)
Get a sorted page of items in a cart, including coupons and promotions.
getCartPromotions(webstoreId, effectiveAccountId, activeCartOrId)
Get promotions for a cart.
getCartPromotions(webstoreId, effectiveAccountId, activeCartOrId, currencyIsoCode)
Get promotions for a cart in a specific currency.
getCartSummary(webstoreId, effectiveAccountId, activeCartOrId)
Get a cart.
getCartSummary(webstoreId, effectiveAccountId, activeCartOrId, currencyIsoCode)
Get a cart in a specific currency.
getOrCreateActiveCartSummary(webstoreId, effectiveAccountId, activeCartOrId)
Get a cart or create an active cart if one doesn’t exist.
getOrCreateActiveCartSummary(webstoreId, effectiveAccountId, activeCartOrId, currencyIsoCode)
Get a cart in a specific currency, or create an active cart if one doesn’t exist.
makeCartPrimary(webstoreId, activeCartOrId, effectiveAccountId)
Make a secondary cart a primary cart.
setCartMessagesVisibility(webstoreId, activeCartOrId, effectiveAccountId, messageVisibility)
Set the visibility for cart messages.
updateCartItem(webstoreId, effectiveAccountId, activeCartOrId, cartItemId, cartItem)
Update an item in a cart.

updateCartItem(webstoreId, effectiveAccountId, activeCartOrId, cartItemId, cartItem, currencyIsoCode)
Update an item in a cart of a specific currency.

upsertInventoryReservation(webstoreId, activeCartOrId, effectiveAccountId, cartInventoryReservationInput) (Pilot)
Create or update an inventory reservation.

addItemToCart(webstoreId, effectiveAccountId, activeCartOrId, cartItemInput, currencyIsoCode)
Add an item to a cart of a specific currency.

API Version
57.0

Available to Guest Users
57.0

Requires Chatter
No

Signature
public static ConnectApi.CartItem addItemToCart(String webstoreId, String effectiveAccountId, String activeCartOrId, ConnectApi.CartItemInput cartItemInput, String currencyIsoCode)

Parameters
webstoreId
Type: String
ID of the webstore.

effectiveAccountId
Type: String
ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.

activeCartOrId
Type: String
ID of the cart, active, or current. The current value indicates a cart with a status that isn’t Closed or PendingDelete.

cartItemInput
Type: ConnectApi.CartItemInput
A ConnectApi.CartItemInput object representing an item to add to the cart.

currencyIsoCode
Type: String
The currency ISO code of the cart.

Return Value
Type: `ConnectApi.CartItem`

Usage
Buyers with read access to carts can add, update, and delete items in carts.
This method respects buyer View Product entitlements and only users entitled to view product data can access it.

```java
addItemsToCart(webstoreId, effectiveAccountId, activeCartOrId, cartItems)
```
Add a batch of up to 100 items to a cart.

API Version
49.0

Available to Guest Users
54.0

Requires Chatter
No

Signature
```java
public static ConnectApi.BatchResult[] addItemsToCart(String webstoreId, String effectiveAccountId, String activeCartOrId, List<ConnectApi.BatchInput> cartItems)
```

Parameters
- `webstoreId`
  Type: `String`
  ID of the webstore.
- `effectiveAccountId`
  Type: `String`
  ID of the buyer account or guest buyer profile for which the request is made. If `null`, the default value is determined from context.
- `activeCartOrId`
  Type: `String`
  ID of the cart, active, or current. The current value is available in version 50.0 and later and indicates a cart with a status that isn’t Closed or PendingDelete.
- `cartItems`
  Type: `List<ConnectApi.BatchInput>`
  The list can contain up to 100 `ConnectApi.BatchInput` objects. In the `ConnectApi.BatchInput` constructor, the input object must be `ConnectApi.CartItemInput`.
Return Value
Type: `ConnectApi.BatchResult[]`

The returned objects correspond to each of the input objects and are returned in the same order as the input objects.
The method call fails only if an error occurs that affects the entire operation (such as a parsing failure). If an individual object causes an error, the error is embedded within the `ConnectApi.BatchResult` list.

Usage
Buyers with read access to carts can add, update, and delete items in carts.
This method respects buyer View Product entitlements and only users entitled to view product data can access it.

`addItemsToCart(webstoreId, effectiveAccountId, activeCartOrId, cartItems, currencyIsoCode)`
Add a batch of up to 100 items to a cart of a specific currency.

API Version
57.0

Available to Guest Users
57.0

Requires Chatter
No

Signature
```java
public static ConnectApi.BatchResult[] addItemsToCart(String webstoreId, String effectiveAccountId, String activeCartOrId, List<ConnectApi.BatchInput> cartItems, String currencyIsoCode)
```

Parameters
- `webstoreId`  
  Type: `String`  
  ID of the webstore.
- `effectiveAccountId`  
  Type: `String`  
  ID of the buyer account or guest buyer profile for which the request is made. If `null`, the default value is determined from context.
- `activeCartOrId`  
  Type: `String`  
  ID of the cart, active, or current. The `current` value is available in version 50.0 and later and indicates a cart with a status that isn't `Closed` or `PendingDelete`.
cartItems
Type: List<ConnectApi.BatchInput>
The list can contain up to 100 ConnectApi.BatchInput objects. In the ConnectApi.BatchInput constructor, the input object must be ConnectApi.CartItemInput.
currencyIsoCode
Type: String
The currency ISO code of the cart.

Return Value
Type: ConnectApi.BatchResult[]
The returned objects correspond to each of the input objects and are returned in the same order as the input objects.
The method call fails only if an error occurs that affects the entire operation (such as a parsing failure). If an individual object causes an error, the error is embedded within the ConnectApi.BatchResult list.

Usage
Buyers with read access to carts can add, update, and delete items in carts.
This method respects buyer View Product entitlements and only users entitled to view product data can access it.

addItemToCart(webstoreId, effectiveAccountId, activeCartOrId, cartItemInput)
Add an item to a cart.

API Version
49.0

Available to Guest Users
54.0

Requires Chatter
No

Signature
public static ConnectApi.CartItem addItemToCart(String webstoreId, String effectiveAccountId, String activeCartOrId, ConnectApi.CartItemInput cartItemInput)

Parameters
webstoreId
Type: String
ID of the webstore.
effectiveAccountId
  Type: String
  ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.

activeCartOrId
  Type: String
  ID of the cart, active, or current. The current value is available in version 50.0 and later and indicates a cart with a status that isn’t Closed or PendingDelete.

cartItemInput
  Type: ConnectApi.CartItemInput
  A ConnectApi.CartItemInput object representing an item to add to the cart.

Return Value
  Type: ConnectApi.CartItem

Usage
Buyers with read access to carts can add, update, and delete items in carts.
This method respects buyer View Product entitlements and only users entitled to view product data can access it.

applyCartCoupon(webstoreId, effectiveAccountId, activeCartOrId, cartCouponInput)
Apply a coupon to a cart.

API Version
54.0

Available to Guest Users
57.0

Requires Chatter
No

Signature
public static ConnectApi.CartCouponCollection applyCartCoupon(String webstoreId, String effectiveAccountId, String activeCartOrId, ConnectApi.cartCouponInput cartCouponInput)

Parameters
webstoreId
  Type: String
  ID of the webstore.
effectiveAccountId
  Type: String
  ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.

activeCartOrId
  Type: String
  ID of the cart, active, or current. The current value is available in version 50.0 and later and indicates a cart with a status that isn't Closed or PendingDelete.

cartCouponInput
  Type: ConnectApi.cartCouponInput
  Coupon code for the coupon.

Return Value
Type: ConnectApi.CartCouponCollection

applyCartCoupon(webstoreId, effectiveAccountId, activeCartOrId, cartCouponInput, currencyIsoCode)
Apply a coupon to a cart.

API Version
57.0

Available to Guest Users
57.0

Requires Chatter
No

Signature
public static ConnectApi.CartCouponCollection applyCartCoupon(String webstoreId, String effectiveAccountId, String activeCartOrId, ConnectApi.cartCouponInput cartCouponInput, String currencyIsoCode)

Parameters
webstoreId
  Type: String
  ID of the webstore.

effectiveAccountId
  Type: String
  ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.
activeCartOrId
  Type: String
  ID of the cart, active, or current. The current value indicates a cart with a status that isn’t Closed or PendingDelete.

cartCouponInput
  Type: ConnectApi.cartCouponInput
  Coupon code for the coupon.

currencyIsoCode
  Type: String
  Currency ISO code of the cart.

Return Value
Type: ConnectApi.CartCouponCollection

copyCartToWishlist(webstoreId, effectiveAccountId, activeCartOrId, cartToWishlistInput)
Copy the products from a cart to a wishlist.

API Version
50.0

Available to Guest Users
54.0

Requires Chatter
No

Signature
public static ConnectApi.CartToWishlistResult copyCartToWishlist(String webstoreId, String effectiveAccountId, String activeCartOrId, ConnectApi.CartToWishlistInput cartToWishlistInput)

Parameters
webstoreId
  Type: String
  ID of the webstore.
effectiveAccountId
  Type: String
  ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.
activeCartOrId
  Type: String
ID of the cart, active, or current. The current value is available in version 50.0 and later and indicates a cart with a status that isn't Closed or PendingDelete.

cartToWishlistInput
  Type: ConnectApi.CartToWishlistInput
  A ConnectApi.CartToWishlistInput object indicating the wishlist to copy products to.

Return Value
Type: ConnectApi.CartToWishlistResult

createCart(webstoreId, cart)
Create a cart.

API Version
49.0

Available to Guest Users
54.0

Requires Chatter
No

Signature
public static ConnectApi.CartSummary createCart(String webstoreId, ConnectApi.CartInput cart)

Parameters
webstoreId
  Type: String
  ID of the webstore.

cart
  Type: ConnectApi.CartInput
  A ConnectApi.CartInput object representing a cart.

Return Value
Type: ConnectApi.CartSummary

Usage
Buyers with read access to carts can create and delete carts.
**deleteCart** (webstoreId, effectiveAccountId, activeCartOrId)

Delete a cart.

**API Version**

49.0

**Available to Guest Users**

54.0

**Requires Chatter**

No

**Signature**

```java
public static Void deleteCart(String webstoreId, String effectiveAccountId, String activeCartOrId)
```

**Parameters**

- **webstoreId**
  - Type: String
  - ID of the webstore.

- **effectiveAccountId**
  - Type: String
  - ID of the buyer account or guest buyer profile for which the request is made. If `null`, the default value is determined from context.

- **activeCartOrId**
  - Type: String
  - ID of the cart, active, or current. The `current` value is available in version 50.0 and later and indicates a cart with a status that isn't `Closed` or `PendingDelete`.

**Return Value**

Type: Void

**Usage**

Buyers with read access to carts can create and delete carts.

**deleteCartCoupon** (webstoreId, effectiveAccountId, activeCartOrId, cartCouponId)

Delete a coupon from a cart.

**API Version**

54.0
Available to Guest Users
57.0

Requires Chatter
No

Signature
public static Void deleteCartCoupon(String webstoreId, String effectiveAccountId, String activeCartOrId, String cartCouponId)

Parameters
webstoreId
Type: String
ID of the webstore.
effectiveAccountId
Type: String
ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.
activeCartOrId
Type: String
ID of the cart, active, or current. The current value is available in version 50.0 and later and indicates a cart with a status that isn’t Closed or PendingDelete.
cartCouponId
Type: String
ID of the cart coupon.

Return Value
Type: Void
deleteCartCoupon(webstoreId, effectiveAccountId, activeCartOrId, cartCouponId, currencyIsoCode)
Delete a coupon from a cart.

API Version
57.0

Available to Guest Users
57.0
Requires Chatter

No

Signature

public static Void deleteCartCoupon(String webstoreId, String effectiveAccountId, String activeCartOrId, String cartCouponId, String currencyIsoCode)

Parameters

webstoreId
Type: String
ID of the webstore.

effectiveAccountId
Type: String
ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.

activeCartOrId
Type: String
ID of the cart, active, or current. The current value indicates a cart with a status that isn’t Closed or PendingDelete.

cartCouponId
Type: String
ID of the cart coupon.

currencyIsoCode
Type: String
Currency ISO code of the cart.

Return Value

Type: Void

deleteCartItem(webstoreId, effectiveAccountId, activeCartOrId, cartItemId)
Delete an item from a cart.

API Version

49.0

Available to Guest Users

54.0

Requires Chatter

No
Signature

```java
public static Void deleteCartItem(String webstoreId, String effectiveAccountId, String activeCartOrId, String cartItemId)
```

Parameters

- `webstoreId`
  - Type: `String`
  - ID of the webstore.

- `effectiveAccountId`
  - Type: `String`
  - ID of the buyer account or guest buyer profile for which the request is made. If `null`, the default value is determined from context.

- `activeCartOrId`
  - Type: `String`
  - ID of the cart, active, or current. The current value is available in version 50.0 and later and indicates a cart with a status that isn’t Closed or PendingDelete.

- `cartItemId`
  - Type: `String`
  - ID of the cart item.

Return Value

Type: `Void`

Usage

Buyers with read access to carts can add, update, and delete items in carts.

```java
delteInventoryReservation(webstoreId, activeCartOrId, effectiveAccountId)
```

( Pilot )

Delete an inventory reservation.

**Note:** This feature is not generally available and is being piloted with certain Customers subject to additional terms and conditions. It is not part of your purchased Services. This feature is subject to change, may be discontinued with no notice at any time in Salesforce’s sole discretion, and Salesforce may never make this feature generally available. Make your purchase decisions only on the basis of generally available products and features. This feature is made available on an AS IS basis and use of this feature is at your sole risk.

API Version

58.0

Available to Guest Users

58.0
Requires Chatter
No

Signature
```
public static Void deleteInventoryReservation(String webstoreId, String activeCartOrId, String effectiveAccountId)
```

Parameters

- **webstoreId**
  Type: String
  ID of the webstore.

- **activeCartOrId**
  Type: String
  ID of the cart, active, or current. The current value indicates a cart with a status that isn't Closed or PendingDelete.

- **effectiveAccountId**
  Type: String
  ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.

Return Value
Type: Void

```
getCartCoupons(webstoreId, effectiveAccountId, activeCartOrId)
```
Get coupons for a cart.

API Version
54.0

Available to Guest Users
57.0

Requires Chatter
No

Signature
```
public static ConnectApi.CartCouponCollection getCartCoupons(String webstoreId, String effectiveAccountId, String activeCartOrId)
```
Parameters

`webstoreId`  
Type: `String`  
ID of the webstore.

`effectiveAccountId`  
Type: `String`  
ID of the buyer account or guest buyer profile for which the request is made. If `null`, the default value is determined from context.

`activeCartOrId`  
Type: `String`  
ID of the cart, active, or current. The current value is available in version 50.0 and later and indicates a cart with a status that isn't Closed or PendingDelete.

Return Value

Type: `ConnectApi.CartCouponCollection`

`getCartCoupons(webstoreId, effectiveAccountId, activeCartOrId, currencyIsoCode)`  
Get coupons for a cart.

API Version

57.0

Available to Guest Users

57.0

Requires Chatter

No

Signature

`public static ConnectApi.CartCouponCollection getCartCoupons(String webstoreId, String effectiveAccountId, String activeCartOrId, String currencyIsoCode)`

Parameters

`webstoreId`  
Type: `String`  
ID of the webstore.

`effectiveAccountId`  
Type: `String`  
ID of the buyer account or guest buyer profile for which the request is made. If `null`, the default value is determined from context.
activeCartOrId
Type: String
ID of the cart, active, or current. The current value indicates a cart with a status that isn’t Closed or PendingDelete.

Return Value
Type: ConnectApi.CartCouponCollection

gGetCartPromotion(webstoreId, effectiveAccountId, activeCartOrId, cartItemPromotionCollectionInput)
Get promotions for a cart item.

API Version
S2.0

Available to Guest Users
S7.0

Requires Chatter
No

Signature
public static ConnectApi.CartItemPromotionCollectionOutputRepresentation getCartItemPromotion(String webstoreId, String effectiveAccountId, String activeCartOrId, ConnectApi.CartItemPromotionCollectionInputRepresentation cartItemPromotionCollectionInput)

Parameters
webstoreId
Type: String
ID of the webstore.
effectiveAccountId
Type: String
ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.
activeCartOrId
Type: String
ID of the cart, active, or current. The current value indicates a cart with a status that isn’t Closed or PendingDelete.
cartItemPromotionCollectionInput
Type: ConnectApi.CartItemPromotionCollectionInputRepresentation
Promotions for a cart item.
Return Value

Type: `ConnectApi.CartItemPromotionCollectionOutputRepresentation`

```java
public static ConnectApi.CartItemPromotionCollectionOutputRepresentation
getCartItemPromotion(String webstoreId, String effectiveAccountId, String activeCartOrId, ConnectApi.CartItemPromotionCollectionInputRepresentation cartItemPromotionCollectionInput, String currencyIsoCode)
```

Get a promotion for a cart item.

API Version

`57.0`

Available to Guest Users

`57.0`

Requires Chatter

No

Signature

```java
public static ConnectApi.CartItemPromotionCollectionOutputRepresentation
getCartItemPromotion(String webstoreId, String effectiveAccountId, String activeCartOrId, ConnectApi.CartItemPromotionCollectionInputRepresentation cartItemPromotionCollectionInput, String currencyIsoCode)
```

Parameters

- `webstoreId`
  Type: `String`
  ID of the webstore.
- `effectiveAccountId`
  Type: `String`
  ID of the buyer account or guest buyer profile for which the request is made. If `null`, the default value is determined from context.
- `activeCartOrId`
  Type: `String`
  ID of the cart, active, or current. The current value indicates a cart with a status that isn’t Closed or PendingDelete.
- `cartItemPromotionCollectionInput`
  Type: `ConnectApi.CartItemPromotionCollectionInputRepresentation`
  Promotions for a cart item.
- `currencyIsoCode`
  Type: `String`
  Currency ISO code of the cart.
Return Value
Type: `ConnectApi.CartItemPromotionCollectionOutputRepresentation`

### getCartItems(webstoreId, effectiveAccountId, activeCartOrId)
Get items in a cart.

**API Version**
49.0

**Available to Guest Users**
54.0

**Requires Chatter**
No

**Signature**
```java
public static ConnectApi.CartItemCollection getCartItems(String webstoreId, String effectiveAccountId, String activeCartOrId)
```

**Parameters**
- `webstoreId`
  Type: `String`
  ID of the webstore.
- `effectiveAccountId`
  Type: `String`
  ID of the buyer account or guest buyer profile for which the request is made. If `null`, the default value is determined from context.
- `activeCartOrId`
  Type: `String`
  ID of the cart, `active`, or `current`. The `current` value is available in version 50.0 and later and indicates a cart with a status that isn’t `Closed` or `PendingDelete`.

**Return Value**
Type: `ConnectApi.CartItemCollection`

### getCartItems(webstoreId, effectiveAccountId, activeCartOrId, pageParam)
Get a page of items in a cart.

**API Version**
49.0
Available to Guest Users
54.0

Requires Chatter
No

Signature
public static ConnectApi.CartItemCollection getCartItems(String webstoreId, String effectiveAccountId, String activeCartOrId, String pageParam)

Parameters
webstoreId
Type: String
ID of the webstore.

effectiveAccountId
Type: String
ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.

activeCartOrId
Type: String
ID of the cart, active, or current. The current value is available in version 50.0 and later and indicates a cart with a status that isn’t Closed or PendingDelete.

pageParam
Type: String
Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in null, the first page is returned.

Return Value
Type: ConnectApi.CartItemCollection

getcartItems(webstoreId, effectiveAccountId, activeCartOrId, pageParam, sortParam)
Get a sorted page of items in a cart.

API Version
49.0

Available to Guest Users
54.0
The function `getCartItems` is used to retrieve a specified size page of items in a cart.

**Parameters**

- **webstoreId**
  - Type: String
  - ID of the webstore.
- **effectiveAccountId**
  - Type: String
  - ID of the buyer account or guest buyer profile for which the request is made. If `null`, the default value is determined from context.
- **activeCartOrId**
  - Type: String
  - ID of the cart, active, or current. The current value is available in version 50.0 and later and indicates a cart with a status that isn’t Closed or PendingDelete.
- **pageParam**
  - Type: String
  - Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.
- **sortParam**
  - Type: `ConnectApi.CartItemSortOrder`
  - Sort order for items in a cart. Values are:
    - CreatedDateAsc—Sorts by oldest creation date.
    - CreatedDateDesc—Sorts by most recent creation date.
    - NameAsc—Sorts by name in ascending alphabetical order (A–Z).
    - NameDesc—Sorts by name in descending alphabetical order (Z–A).
    - SalesPriceAsc—Sorts from lowest to highest negotiated price.
    - SalesPriceDesc—Sorts from highest to lowest negotiated price.
  - If `null`, the default is CreatedDateDesc.

**Return Value**

- Type: `ConnectApi.CartItemCollection`
  - `getCartItems(webstoreId, effectiveAccountId, activeCartOrId, pageParam, pageSize)`

Get a specified size page of items in a cart.
API Version
49.0

Available to Guest Users
54.0

Requires Chatter
No

Signature
public static ConnectApi.CartItemCollection getCartItems(String webstoreId, String effectiveAccountId, String activeCartOrId, String pageParam, Integer pageSize)

Parameters
webstoreId
Type: String
ID of the webstore.

effectiveAccountId
Type: String
ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.

activeCartOrId
Type: String
ID of the cart, active, or current. The current value is available in version 50.0 and later and indicates a cart with a status that isn’t Closed or PendingDelete.

pageParam
Type: String
Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

Return Value
Type: ConnectApi.CartItemCollection

getcartItems(webstoreId, effectiveAccountId, activeCartOrId, pageParam, pageSize, sortParam)
Get a specified size, sorted page of items in a cart.
API Version
49.0

Available to Guest Users
54.0

Requires Chatter
No

Signature

```java
public static ConnectApi.CartItemCollection getCartItems(String webstoreId, String effectiveAccountId, String activeCartOrId, String pageParam, Integer pageSize, ConnectApi.CartItemSortOrder sortParam)
```

Parameters

- `webstoreId`
  Type: `String`
  ID of the webstore.

- `effectiveAccountId`
  Type: `String`
  ID of the buyer account or guest buyer profile for which the request is made. If `null`, the default value is determined from context.

- `activeCartOrId`
  Type: `String`
  ID of the cart, active, or current. The current value is available in version 50.0 and later and indicates a cart with a status that isn’t Closed or PendingDelete.

- `pageParam`
  Type: `String`
  Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

- `pageSize`
  Type: `Integer`
  Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

- `sortParam`
  Type: `ConnectApi.CartItemSortOrder`
  Sort order for items in a cart. Values are:
  - `CreatedDateAsc`—Sorts by oldest creation date.
  - `CreatedDateDesc`—Sorts by most recent creation date.
  - `NameAsc`—Sorts by name in ascending alphabetical order (A–Z).
  - `NameDesc`—Sorts by name in descending alphabetical order (Z–A).
  - `SalesPriceAsc`—Sorts from lowest to highest negotiated price.
• **SalesPriceDesc**—Sorts from highest to lowest negotiated price.
  If `null`, the default is `CreatedDateDesc`.

**Return Value**
Type: `ConnectApi.CartItemCollection`

```java
public static ConnectApi.CartItemCollection getCartItems(String webstoreId, String effectiveAccountId, String activeCartOrId, String productFields, String pageParam, Integer pageSize, ConnectApi.CartItemSortOrder sortParam)
```

Get a specified size, sorted page of items filtered by product fields in a cart.

**API Version**
49.0

**Available to Guest Users**
54.0

**Requires Chatter**
No

**Signature**
```java
public static ConnectApi.CartItemCollection getCartItems(String webstoreId, String effectiveAccountId, String activeCartOrId, String productFields, String pageParam, Integer pageSize, ConnectApi.CartItemSortOrder sortParam)
```

**Parameters**
- **webstoreId**
  Type: `String`
  ID of the webstore.
- **effectiveAccountId**
  Type: `String`
  ID of the buyer account or guest buyer profile for which the request is made. If `null`, the default value is determined from context.
- **activeCartOrId**
  Type: `String`
  ID of the cart, active, or current. The `current` value is available in version 50.0 and later and indicates a cart with a status that isn’t `Closed` or `PendingDelete`.
- **productFields**
  Type: `String`
  Comma-separated list of up to five product fields. Results include fields that you have access to. Some product fields (such as `productName` and `sku`) are returned even when not included in the `productFields` parameter.
pageParam
  Type: String
  Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in `null`, the first page is returned.

pageSize
  Type: Integer
  Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

sortParam
  Type: ConnectApi.CartItemSortOrder
  Sort order for items in a cart. Values are:
  - CreatedDateAsc—Sorts by oldest creation date.
  - CreatedDateDesc—Sorts by most recent creation date.
  - NameAsc—Sorts by name in ascending alphabetical order (A–Z).
  - NameDesc—Sorts by name in descending alphabetical order (Z–A).
  - SalesPriceAsc—Sorts from lowest to highest negotiated price.
  - SalesPriceDesc—Sorts from highest to lowest negotiated price.
  If `null`, the default is CreatedDateDesc.

Return Value
  Type: ConnectApi.CartItemCollection

getCartItems(webstoreId, effectiveAccountId, activeCartOrId, productFields, pageParam, pageSize, sortParam, currencyIsoCode)
Get a specified size, sorted page of items filtered by product fields in a cart.

API Version
57.0

Available to Guest Users
57.0

Requires Chatter
No

Signature
public static ConnectApi.CartItemCollection getCartItems(String webstoreId, String effectiveAccountId, String activeCartOrId, String productFields, String pageParam, Integer pageSize, ConnectApi.CartItemSortOrder sortParam, String currencyIsoCode)
Parameters

webstoreId
Type: String
ID of the webstore.

effectiveAccountId
Type: String
ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.

activeCartOrId
Type: String
ID of the cart, active, or current. The current value is available in version 50.0 and later and indicates a cart with a status that isn't Closed or PendingDelete.

productFields
Type: String
Comma-separated list of up to five product fields. Results include fields that you have access to. Some product fields (such as productName and sku) are returned even when not included in the productFields parameter.

pageParam
Type: String
Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

currencyIsoCode
Type: String
Currency ISO code of the cart.

Return Value
Type: ConnectApi.CartItemCollection

gGetCartItems(webstoreId, effectiveAccountId, activeCartOrId, productFields, pageParam, pageSize, sortParam, currencyIsoCode, includePromotions, includeCoupons)
Get a sorted page of items in a cart, including coupons and promotions.

API Version
59.0

Available to Guest Users
54.0
Requires Chatter
No

Signature

```java
public static ConnectApi.CartItemCollection getCartItems(String webstoreId, String effectiveAccountId, String activeCartOrId, String productFields, String pageParam, Integer pageSize, ConnectApi.CartItemSortOrder sortParam, String currencyIsoCode, Boolean includePromotions, Boolean includeCoupons)
```

Parameters

- `webstoreId`
  Type: `String`
  ID of the webstore.

- `effectiveAccountId`
  Type: `String`
  ID of the buyer account or guest buyer profile for which the request is made. If `null`, the default value is determined from context.

- `activeCartOrId`
  Type: `String`
  ID of the cart, active, or current. The `current` value is available in version 50.0 and later and indicates a cart with a status that isn’t Closed or PendingDelete.

- `productFields`
  Type: `String`
  Comma-separated list of up to five product fields. Results include fields that you have access to. Some product fields (such as `productName` and `sku`) are returned even when not included in the `productFields` parameter.

- `pageParam`
  Type: `String`
  Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

- `pageSize`
  Type: `Integer`
  Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

- `includePromotions`
  Type: `Boolean`
  Indicates whether to include coupons (`True`) or not (`False`)

- `includeCoupons`
  Type: `Boolean`
  Indicates whether to include promotions (`True`) or not (`False`).

Return Value

Type: `ConnectApi.CartItemCollection`
getCartPromotions(webstoreId, effectiveAccountId, activeCartOrId)

Get promotions for a cart.

API Version
53.0

Available to Guest Users
57.0

Requires Chatter
No

Signature
public static ConnectApi.CartPromotionCollection getCartPromotions(String webstoreId, String effectiveAccountId, String activeCartOrId)

Parameters
webstoreId
  Type: String
  ID of the webstore.

effectiveAccountId
  Type: String
  ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.

activeCartOrId
  Type: String
  ID of the cart, active, or current. The current value indicates a cart with a status that isn’t Closed or PendingDelete.

Return Value
Type: ConnectApi.CartPromotionCollection

getCodePromotions(webstoreId, effectiveAccountId, activeCartOrId, currencyIsoCode)

Get promotions for a cart in a specific currency.

API Version
57.0

Available to Guest Users
57.0
Requires Chatter
No

Signature

```java
public static ConnectApi.CartPromotionCollection getCartPromotions(String webstoreId, String effectiveAccountId, String activeCartOrId, String currencyIsoCode)
```

Parameters

- **webstoreId**
  Type: String
  ID of the webstore.

- **effectiveAccountId**
  Type: String
  ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.

- **activeCartOrId**
  Type: String
  ID of the cart, active, or current. The current value indicates a cart with a status that isn’t Closed or PendingDelete.

- **currencyIsoCode**
  Type: String
  Currency ISO code of the cart.

Return Value

Type: ConnectApi.CartPromotionCollection

getCartSummary(webstoreId, effectiveAccountId, activeCartOrId)

Get a cart.

API Version

49.0

Available to Guest Users

54.0

Requires Chatter

No

Signature

```java
public static ConnectApi.CartSummary getCartSummary(String webstoreId, String effectiveAccountId, String activeCartOrId)
```
Parameters

webstoreId
Type: String
ID of the webstore.

effectiveAccountId
Type: String
ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.

activeCartOrId
Type: String
ID of the cart, active, or current. The current value is available in version 50.0 and later and indicates a cart with a status that isn’t Closed or PendingDelete. If you specify active and there isn’t an active cart, you get an error.

Return Value
Type: ConnectApi.CartSummary

gGetCartSummary(webstoreId, effectiveAccountId, activeCartOrId, currencyIsoCode)
Get a cart in a specific currency.

API Version
57.0

Available to Guest Users
57.0

Requires Chatter
No

Signature

public static ConnectApi.CartSummary getCartSummary(String webstoreId, String effectiveAccountId, String activeCartOrId, String currencyIsoCode)

Parameters

webstoreId
Type: String
ID of the webstore.

effectiveAccountId
Type: String
ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.
activeCartOrId
Type: String
ID of the cart, active, or current. The current value is available in version 50.0 and later and indicates a cart with a status that isn’t Closed or PendingDelete. If you specify active and there isn’t an active cart, you get an error.

currencyIsoCode
Type: String
Currency ISO code of the cart.

Return Value
Type: ConnectApi.CartSummary

getOrCreateActiveCartSummary(webstoreId, effectiveAccountId, activeCartOrId)
Get a cart or create an active cart if one doesn’t exist.

API Version
49.0

Available to Guest Users
54.0

Requires Chatter
No

Signature
public static ConnectApi.CartSummary getOrCreateActiveCartSummary(String webstoreId, String effectiveAccountId, String activeCartOrId)

Parameters

webstoreId
Type: String
ID of the webstore.

effectiveAccountId
Type: String
ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.

activeCartOrId
Type: String
ID of the cart, active, or current. The current value is available in version 50.0 and later and indicates a cart with a status that isn’t Closed or PendingDelete. If you specify active and there isn’t an active cart, one is created.
Return Value
Type: `ConnectApi.CartSummary`

Usage
Buyers with read access to carts can create and delete carts.

`getOrCreateActiveCartSummary(webstoreId, effectiveAccountId, activeCartOrId, currencyIsoCode)`
Get a cart in a specific currency, or create an active cart if one doesn’t exist.

API Version
57.0

Available to Guest Users
57.0

Requires Chatter
No

Signature
`public static ConnectApi.CartSummary getOrCreateActiveCartSummary(String webstoreId, String effectiveAccountId, String activeCartOrId, String currencyIsoCode)`

Parameters
`webstoreId`
Type: `String`
ID of the webstore.
`effectiveAccountId`
Type: `String`
ID of the buyer account or guest buyer profile for which the request is made. If `null`, the default value is determined from context.
`activeCartOrId`
Type: `String`
ID of the cart, `active`, or `current`. The `current` value is available in version 50.0 and later and indicates a cart with a status that isn’t `Closed` or `PendingDelete`. If you specify `active` and there isn’t an active cart, one is created.
`currencyIsoCode`
Type: `String`
Currency ISO code of the cart.
Return Value
Type: `ConnectApi.CartSummary`

Usage
Buyers with read access to carts can create and delete carts.

`makeCartPrimary(webstoreId, activeCartOrId, effectiveAccountId)`
Make a secondary cart a primary cart.

API Version
53.0

Available to Guest Users
56.0

Requires Chatter
Yes

Signature
```java
public static ConnectApi.CommerceActionResult makeCartPrimary(String webstoreId, String activeCartOrId, String effectiveAccountId)
```

Parameters
- `webstoreId`
  Type: `String`
  ID of the webstore.
- `activeCartOrId`
  Type: `String`
  ID of the cart, active, or current. The current value is available in version 50.0 and later and indicates a cart with a status that isn't Closed or PendingDelete.
- `effectiveAccountId`
  Type: `String`
  ID of the buyer account or guest buyer profile for which the request is made. If `null`, the default value is determined from context.

Return Value
Type: `ConnectApi.CommerceActionResult`
setCartMessagesVisibility(webstoreId, activeCartOrId, effectiveAccountId, messageVisibility)

Set the visibility for cart messages.

API Version
50.0

Available to Guest Users
54.0

Requires Chatter
No

Signature
public static ConnectApi.CartMessagesVisibilityResult setCartMessagesVisibility(String webstoreId, String activeCartOrId, String effectiveAccountId, ConnectApi.CartMessagesVisibilityInput messageVisibility)

Parameters
webstoreId
  Type: String
  ID of the webstore.
activeCartOrId
  Type: String
  ID of the cart, active, or current. The current value is available in version 50.0 and later and indicates a cart with a status that isn’t Closed or PendingDelete.
effectiveAccountId
  Type: String
  ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.
messageVisibility
  Type: ConnectApi.CartMessagesVisibilityInput
  A ConnectApi.CartMessagesVisibilityInput object specifying the visibility setting.

Return Value
Type: ConnectApi.CartMessagesVisibilityResult

updateCartItem(webstoreId, effectiveAccountId, activeCartOrId, cartItemId, cartItem)

Update an item in a cart.
API Version

49.0

Available to Guest Users

54.0

Requires Chatter

No

Signature

public static ConnectApi.CartItem updateCartItem(String webstoreId, String effectiveAccountId, String activeCartOrId, String cartItemId, ConnectApi.CartItemInput cartItem)

Parameters

webstoreId
  Type: String
  ID of the webstore.

effectiveAccountId
  Type: String
  ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.

activeCartOrId
  Type: String
  ID of the cart, active, or current. The current value is available in version 50.0 and later and indicates a cart with a status that isn’t Closed or PendingDelete.

cartItemId
  Type: String
  ID of the cart item.

cartItem
  Type: ConnectApi.CartItemInput
  A ConnectApi.CartItemInput object representing a cart item to update.

Return Value

Type: ConnectApi.CartItem

Usage

Buyers with read access to carts can add, update, and delete items in carts.

This method respects buyer View Product entitlements and only users entitled to view product data can access it.
updateCartItem(webstoreId, effectiveAccountId, activeCartOrId, cartItemId, cartItem, currencyIsoCode)

Update an item in a cart of a specific currency.

API Version
57.0

Available to Guest Users
57.0

Requires Chatter
No

Signature
public static ConnectApi.CartItem updateCartItem(String webstoreId, String effectiveAccountId, String activeCartOrId, String cartItemId, ConnectApi.CartItemInput cartItem, String currencyIsoCode)

Parameters

webstoreId
  Type: String
  ID of the webstore.

effectiveAccountId
  Type: String
  ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.

activeCartOrId
  Type: String
  ID of the cart, active, or current. The current value is available in version 50.0 and later and indicates a cart with a status that isn't Closed or PendingDelete.

cartItemId
  Type: String
  ID of the cart item.

cartItem
  Type: ConnectApi.CartItemInput
  A ConnectApi.CartItemInput object representing a cart item to update.

currencyIsoCode
  Type: String
  The currency ISO code of the cart.
Return Value
Type: `ConnectApi.CartItem`

Usage
Buyers with read access to carts can add, update, and delete items in carts.
This method respects buyer View Product entitlements and only users entitled to view product data can access it.

```java
upsertInventoryReservation(webstoreId, activeCartOrId, effectiveAccountId, cartInventoryReservationInput) (Pilot)
```
Create or update an inventory reservation.

⚠️ **Note:** This feature is not generally available and is being piloted with certain Customers subject to additional terms and conditions. It is not part of your purchased Services. This feature is subject to change, may be discontinued with no notice at any time in Salesforce's sole discretion, and Salesforce may never make this feature generally available. Make your purchase decisions only on the basis of generally available products and features. This feature is made available on an AS IS basis and use of this feature is at your sole risk.

API Version
58.0

Available to Guest Users
58.0

Requires Chatter
No

Signature
```java
public static ConnectApi.CartInventoryReservationOutputRepresentation upsertInventoryReservation(String webstoreId, String activeCartOrId, String effectiveAccountId, ConnectApi.CartInventoryReservationInputRepresentation cartInventoryReservationInput)
```

Parameters
- `webstoreId`
  Type: `String`
  ID of the webstore.
- `activeCartOrId`
  Type: `String`
  ID of the cart, active, or current. The current value indicates a cart with a status that isn't Closed or PendingDelete.
- `effectiveAccountId`
  Type: `String`
ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.

**cartInventoryReservationInput**

Type: `ConnectApi.CartInventoryReservationInputRepresentation`

A ConnectApi.CartInventoryReservationInputRepresentation input class indicating the reservation duration.

**Return Value**

Type: `ConnectApi.CartInventoryReservationOutputRepresentation`

### CommerceCatalog Class

Get products, product categories, and product category paths.

### Namespace

`ConnectApi`

### CommerceCatalog Methods

These methods are for CommerceCatalog. All methods are static.

**IN THIS SECTION:**

- `getProduct(webstoreId, productId, effectiveAccountId, fields, excludeFields, mediaGroups, excludeMedia, excludeEntitlementDetails, excludePrimaryProductCategory)`
  - Get a product.
- `getProduct(webstoreId, productId, effectiveAccountId, fields, excludeFields, mediaGroups, excludeMedia, excludeEntitlementDetails, excludePrimaryProductCategory, excludeVariationInfo, excludeAttributeSetInfo)`
  - Get a product with variation and attribute information.
- `getProduct(webstoreId, productId, effectiveAccountId, fields, excludeFields, mediaGroups, excludeMedia, excludeEntitlementDetails, excludePrimaryProductCategory, excludeVariationInfo, excludeAttributeSetInfo, excludeQuantityRule)`
  - Get a product with quantity rule information.
- `getProduct(webstoreId, productId, effectiveAccountId, fields, excludeFields, mediaGroups, excludeMedia, excludeEntitlementDetails, excludePrimaryProductCategory, excludeVariationInfo, excludeAttributeSetInfo, excludeQuantityRule, excludeProductSellingModels)`
  - Get detailed information for a product, optionally including information about its product selling models.
- `getProduct(webstoreId, productId, effectiveAccountId, fields, mediaGroups, excludeFields, excludeMedia, excludeEntitlementDetails, excludePrimaryProductCategory, excludeVariationInfo, excludeAttributeSetInfo, excludeQuantityRule, excludeProductSellingModels)`
  - Get detailed information for a product without its entitlement information.
- `getProductCategory(webstoreId, productCategoryId, effectiveAccountId, fields, excludeFields, mediaGroups, excludeMedia)`
  - Get a product category.
- `getProductCategoryChildren(webstoreId, effectiveAccountId, parentProductCategoryId, fields, excludeFields, mediaGroups, excludeMedia)`
  - Get product categories.
- `getProductCategoryPath(webstoreId, productCategoryId)`
  - Get the product category path from the root category to the current category.
getProductChildCollection(webstoreId, productId, effectiveAccountId, fields, mediaGroups, excludeFields, excludeMedia, excludeAttributeSetInfo, excludeQuantityRule, pageToken, pageSize)
Get a collection of child products related to a parent product.

getProducts(webstoreId, effectiveAccountId, ids, skus, fields, excludeMedia, excludePrices)
Get fields, prices, and default images for a list of products.

getProducts(webstoreId, effectiveAccountId, ids, skus, fields, excludeMedia)
Get fields and default images for a list of products.

getProduct(webstoreId, productId, effectiveAccountId, fields, excludeFields, mediaGroups, excludeMedia, excludeEntitlementDetails, excludePrimaryProductCategory)
Get a product.

API Version
49.0

Available to Guest Users
51.0

Requires Chatter
No

Signature
public static ConnectApi.ProductDetail getProduct(String webstoreId, String productId, String effectiveAccountId, List<String> fields, Boolean excludeFields, List<String> mediaGroups, Boolean excludeMedia, Boolean excludeEntitlementDetails, Boolean excludePrimaryProductCategory)

Parameters

webstoreId
Type: String
ID of the webstore.

productId
Type: String
ID of the product.

effectiveAccountId
Type: String
ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.

fields
Type: List<String>
Comma-separated list of field names.

If this list is empty or unspecified, all fields are returned. There is no limit to the number of fields you can specify. The number of fields and number of characters in the field name may affect the URL size limit. If excludeFields and fields are specified, the excludeFields parameter takes precedence.

**excludeFields**

Type: Boolean

Specifies whether the fields are returned (false) or not (true). If unspecified, defaults to false.

**mediaGroups**

Type: List<String>

Comma-separated list of developer names of media group records.

If this list is empty or unspecified, all media groups are returned. If excludeMedia and mediaGroups are specified, the excludeMedia parameter takes precedence.

**excludeMedia**

Type: Boolean

Specifies whether the media groups and default images of the product are returned (false) or not (true). If unspecified, defaults to false.

**excludeEntitlementDetails**

Type: Boolean

Specifies whether the entitlement details of the product are returned (false) or not (true). If unspecified, defaults to false.

**excludePrimaryProductCategory**

Type: Boolean

Specifies whether the primary category path of the product is returned (false) or not (true). If unspecified, defaults to false.

**Return Value**

Type: ConnectApi.ProductDetail

**Usage**

This method respects buyer View Product entitlements and only users entitled to view product data can access it.

```java
getProduct(webstoreId, productId, effectiveAccountId, fields, excludeFields, mediaGroups, excludeMedia, excludeEntitlementDetails, excludePrimaryProductCategory, excludeVariationInfo, excludeAttributeSetInfo)
```

Get a product with variation and attribute information.

**API Version**

50.0

**Available to Guest Users**

51.0
Requires Chatter
No

Signature

```java
public static ConnectApi.ProductDetail getProduct(String webstoreId, String productId,
String effectiveAccountId, List<String> fields, Boolean excludeFields, List<String>
mediaGroups, Boolean excludeMedia, Boolean excludeEntitlementDetails, Boolean excludePrimaryProductCategory, Boolean excludeVariationInfo, Boolean excludeAttributeSetInfo)
```

Parameters

`webstoreId`
Type: String
ID of the webstore.

`productId`
Type: String
ID of the product.

`effectiveAccountId`
Type: String
ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.

`fields`
Type: List<String>
Comma-separated list of field names.
If this list is empty or unspecified, all fields are returned. There is no limit to the number of fields you can specify. The number of fields and number of characters in the field name may affect the URL size limit. If excludeFields and fields are specified, the excludeFields parameter takes precedence.

`excludeFields`
Type: Boolean
Specifies whether the fields are returned (false) or not (true). If unspecified, defaults to false.

`mediaGroups`
Type: List<String>
Comma-separated list of developer names of media group records.
If this list is empty or unspecified, all media groups are returned. If excludeMedia and mediaGroups are specified, the excludeMedia parameter takes precedence.

`excludeMedia`
Type: Boolean
Specifies whether the media groups and default images of the product are returned (false) or not (true). If unspecified, defaults to false.

`excludeEntitlementDetails`
Type: Boolean
Specifies whether the entitlement details of the product are returned (false) or not (true). If unspecified, defaults to false.
excludePrimaryProductCategory
Type: Boolean
Specifies whether the primary category path of the product is returned (false) or not (true). If unspecified, defaults to false.

excludeVariationInfo
Type: Boolean
Specifies whether the variation information for the product is returned (false) or not (true). If unspecified, defaults to false.

excludeAttributeSetInfo
Type: Boolean
Specifies whether the attribute set information for the product is returned (false) or not (true). If unspecified, defaults to false.

Return Value
Type: ConnectApi.ProductDetail

Usage
This method respects buyer View Product entitlements and only users entitled to view product data can access it.

getProduct(webstoreId, productId, effectiveAccountId, fields, excludeFields, mediaGroups, excludeMedia, excludeEntitlementDetails, excludePrimaryProductCategory, excludeVariationInfo, excludeAttributeSetInfo, excludeQuantityRule)
Get a product with quantity rule information.

API Version
52.0

Available to Guest Users
52.0

Requires Chatter
No

Signature
public static ConnectApi.ProductDetail getProduct(String webstoreId, String productId, String effectiveAccountId, List<String> fields, Boolean excludeFields, List<String> mediaGroups, Boolean excludeMedia, Boolean excludeEntitlementDetails, Boolean excludePrimaryProductCategory, Boolean excludeVariationInfo, Boolean excludeAttributeSetInfo, Boolean excludeQuantityRule)
Parameters

`webstoreId`
Type: String
ID of the webstore.

`productId`
Type: String
ID of the product.

`effectiveAccountId`
Type: String
ID of the buyer account or guest buyer profile for which the request is made. If `null`, the default value is determined from context.

`fields`
Type: List<String>
Comma-separated list of field names.

If this list is empty or unspecified, all fields are returned. There is no limit to the number of fields you can specify. The number of fields and number of characters in the field name may affect the URL size limit. If `excludeFields` and `fields` are specified, the `excludeFields` parameter takes precedence.

`excludeFields`
Type: Boolean
Specifies whether the fields are returned (`false`) or not (`true`). If unspecified, defaults to `false`.

`mediaGroups`
Type: List<String>
Comma-separated list of developer names of media group records.

If this list is empty or unspecified, all media groups are returned. If `excludeMedia` and `mediaGroups` are specified, the `excludeMedia` parameter takes precedence.

`excludeMedia`
Type: Boolean
Specifies whether the media groups and default images of the product are returned (`false`) or not (`true`). If unspecified, defaults to `false`.

`excludeEntitlementDetails`
Type: Boolean
Specifies whether the entitlement details of the product are returned (`false`) or not (`true`). If unspecified, defaults to `false`.

`excludePrimaryProductCategory`
Type: Boolean
Specifies whether the primary category path of the product is returned (`false`) or not (`true`). If unspecified, defaults to `false`.

`excludeVariationInfo`
Type: Boolean
Specifies whether the variation information for the product is returned (`false`) or not (`true`). If unspecified, defaults to `false`.

`excludeAttributeSetInfo`
Type: Boolean
Specifies whether the attribute set information for the product is returned (`false`) or not (`true`). If unspecified, defaults to `false`.
excludeQuantityRule
Type: Boolean
Specifies whether the quantity rule information for the product is returned (false) or not (true). If unspecified, defaults to false.

Return Value
Type: ConnectApi.ProductDetail

Usage
This method respects buyer View Product entitlements and only users entitled to view product data can access it.

getProduct(webstoreId, productId, effectiveAccountId, fields, excludeFields, mediaGroups, excludeMedia, excludeEntitlementDetails, excludePrimaryProductCategory, excludeVariationInfo, excludeAttributeSetInfo, excludeQuantityRule, excludeProductSellingModels)
Get detailed information for a product, optionally including information about its product selling models.

API Version
56.0

Available to Guest Users
56.0

Requires Chatter
No

Signature
public static ConnectApi.ProductDetail getProduct(String webstoreId, String productId, String effectiveAccountId, List<String> fields, Boolean excludeFields, List<String> mediaGroups, Boolean excludeMedia, Boolean excludeEntitlementDetails, Boolean excludePrimaryProductCategory, Boolean excludeVariationInfo, Boolean excludeAttributeSetInfo, Boolean excludeQuantityRule, Boolean excludeProductSellingModels)

Parameters
webstoreId
Type: String
ID of the webstore.

productId
Type: String
ID of the product.
effectiveAccountId
   Type: String
   ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.

fields
   Type: List<String>
   Comma-separated list of field names.
   If this list is empty or unspecified, all fields are returned. There is no limit to the number of fields you can specify. The number of fields and number of characters in the field name may affect the URL size limit. If excludeFields and fields are specified, the excludeFields parameter takes precedence.

excludeFields
   Type: Boolean
   Specifies whether the fields are returned (false) or not (true). If unspecified, defaults to false.

mediaGroups
   Type: List<String>
   Comma-separated list of developer names of media group records.
   If this list is empty or unspecified, all media groups are returned. If excludeMedia and mediaGroups are specified, the excludeMedia parameter takes precedence.

excludeMedia
   Type: Boolean
   Specifies whether the media groups and default images of the product are returned (false) or not (true). If unspecified, defaults to false.

excludeEntitlementDetails
   Type: Boolean
   Specifies whether the entitlement details of the product are returned (false) or not (true). If unspecified, defaults to false.

excludePrimaryProductCategory
   Type: Boolean
   Specifies whether the primary category path of the product is returned (false) or not (true). If unspecified, defaults to false.

excludeVariationInfo
   Type: Boolean
   Specifies whether the variation information for the product is returned (false) or not (true). If unspecified, defaults to false.

excludeAttributeSetInfo
   Type: Boolean
   Specifies whether the attribute set information for the product is returned (false) or not (true). If unspecified, defaults to false.

excludeQuantityRule
   Type: Boolean
   Specifies whether the quantity rule information for the product is returned (false) or not (true). If unspecified, defaults to false.

excludeProductSellingModels
   Type: Boolean
   Specifies whether product selling models are returned or not. The behavior of this parameter depends on whether you turn on the CommerceSubscription permission. If the permission is on, and if you set the parameter to false (or if you omit the parameter), product selling models are returned. If the permission is on, and if you set the parameter to true, product selling models are not
returned. If the permission is off, product selling models are not returned, regardless of whether you omit the parameter or provide a value.

**Return Value**
Type: `ConnectApi.ProductDetail`

**Usage**
This method respects buyer View Product entitlements and only users entitled to view product data can access it.

```java
getProduct(String webstoreId, String productId, String effectiveAccountId, List<String> fields, List<String> mediaGroups, Boolean excludeFields, Boolean excludeMedia, Boolean excludePrimaryProductCategory, Boolean excludeVariationInfo, Boolean excludeAttributeSetInfo, Boolean excludeQuantityRule, Boolean excludeProductSellingModels)
```
Get detailed information for a product without its entitlement information.

**API Version**
57.0

**Available to Guest Users**
57.0

**Requires Chatter**
No

**Signature**
```java
public static ConnectApi.ProductDetail getProduct(String webstoreId, String productId, String effectiveAccountId, List<String> fields, List<String> mediaGroups, Boolean excludeFields, Boolean excludeMedia, Boolean excludePrimaryProductCategory, Boolean excludeVariationInfo, Boolean excludeAttributeSetInfo, Boolean excludeQuantityRule, Boolean excludeProductSellingModels)
```

**Parameters**
- `webstoreId`
  Type: `String`
  ID of the webstore.
- `productId`
  Type: `String`
  ID of the product.
- `effectiveAccountId`
  Type: `String`
ID of the buyer account or guest buyer profile for which the request is made. If `null`, the default value is determined from context.

**fields**
- **Type**: `List<String>`
- Comma-separated list of field names.
  - If this list is empty or unspecified, all fields are returned. There is no limit to the number of fields you can specify. The number of fields and number of characters in the field name may affect the URL size limit. If `excludeFields` and `fields` are specified, the `excludeFields` parameter takes precedence.

**mediaGroups**
- **Type**: `List<String>`
- Comma-separated list of developer names of media group records.
  - If this list is empty or unspecified, all media groups are returned. If `excludeMedia` and `mediaGroups` are specified, the `excludeMedia` parameter takes precedence.

**excludeFields**
- **Type**: `Boolean`
- Specifies whether the fields are returned (`false`) or not (`true`). If unspecified, defaults to `false`.

**excludeMedia**
- **Type**: `Boolean`
- Specifies whether the media groups and default images of the product are returned (`false`) or not (`true`). If unspecified, defaults to `false`.

**excludePrimaryProductCategory**
- **Type**: `Boolean`
- Specifies whether the primary category path of the product is returned (`false`) or not (`true`). If unspecified, defaults to `false`.

**excludeVariationInfo**
- **Type**: `Boolean`
- Specifies whether the variation information for the product is returned (`false`) or not (`true`). If unspecified, defaults to `false`.

**excludeAttributeSetInfo**
- **Type**: `Boolean`
- Specifies whether the attribute set information for the product is returned (`false`) or not (`true`). If unspecified, defaults to `false`.

**excludeQuantityRule**
- **Type**: `Boolean`
- Specifies whether the quantity rule information for the product is returned (`false`) or not (`true`). If unspecified, defaults to `false`.

**excludeProductSellingModels**
- **Type**: `Boolean`
- Specifies whether product selling models are returned or not. The behavior of this parameter depends on whether you turn on the CommerceSubscription permission. If the permission is on, and if you set the parameter to `false` (or if you omit the parameter), product selling models are returned. If the permission is on, and if you set the parameter to `true`, product selling models are not returned. If the permission is off, product selling models are not returned, regardless of whether you omit the parameter or provide a value.

**Return Value**
- **Type**: `ConnectApi.ProductDetail`
Usage

This method respects buyer View Product entitlements and only users entitled to view product data can access it.

**getProductCategory(webstoreId, productCategoryId, effectiveAccountId, fields, excludeFields, mediaGroups, excludeMedia)**

Get a product category.

API Version

49.0

Available to Guest Users

51.0

Requires Chatter

No

Signature

```java
public static ConnectApi.ProductCategoryDetail getProductCategory(String webstoreId, String productCategoryId, String effectiveAccountId, List<String> fields, Boolean excludeFields, List<String> mediaGroups, Boolean excludeMedia)
```

Parameters

- **webstoreId**
  - Type: String
  - ID of the webstore.

- **productCategoryId**
  - Type: String
  - ID of the product category.

- **effectiveAccountId**
  - Type: String
  - ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.

- **fields**
  - Type: List<String>
  - Comma-separated list of field names.

  If this list is empty or unspecified, all fields are returned. There is no limit to the number of fields you can specify. The number of fields and number of characters in the field name may affect the URL size limit. If excludeFields and fields are specified, the excludeFields parameter takes precedence.

- **excludeFields**
  - Type: Boolean
  - Specifies whether the fields are returned (false) or not (true). If unspecified, defaults to false.
mediaGroups
Type: List<String>
Comma-separated list of developer names of media group records.
If this list is empty or unspecified, all media groups are returned. If excludeMedia and mediaGroups are specified, the
excludeMedia parameter takes precedence.

excludeMedia
Type: Boolean
Specifies whether the media groups and default images of the product are returned (false) or not (true). If unspecified, defaults to false.

Return Value
Type: ConnectApi.ProductCategoryDetail

getProductCategoryChildren(webstoreId, effectiveAccountId, parentProductCategoryId, fields, excludeFields, mediaGroups, excludeMedia)
Get product categories.

API Version
52.0

Available to Guest Users
52.0

Requires Chatter
No

Signature
public static ConnectApi.ProductCategoryDetailCollection
getProductCategoryChildren(String webstoreId, String effectiveAccountId, String parentProductCategoryId, List<String> fields, Boolean excludeFields, List<String> mediaGroups, Boolean excludeMedia)

Parameters
webstoreId
Type: String
ID of the webstore.
effectiveAccountId
Type: String
ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.
**parentProductCategoryId**
Type: String
ID of the product category for which you want to get all the children product categories. If `null`, returns all the top-level product categories for the store.

**fields**
Type: List<String>
Comma-separated list of field names.

If this list is empty or unspecified, all fields are returned. There is no limit to the number of fields you can specify. The number of fields and number of characters in the field name may affect the URL size limit. If `excludeFields` and `fields` are specified, the `excludeFields` parameter takes precedence.

**excludeFields**
Type: Boolean
Specifies whether the fields are returned (`false`) or not (`true`). If unspecified, defaults to `false`.

**mediaGroups**
Type: List<String>
Comma-separated list of developer names of media group records.

If this list is empty or unspecified, all media groups are returned. If `excludeMedia` and `mediaGroups` are specified, the `excludeMedia` parameter takes precedence.

**excludeMedia**
Type: Boolean
Specifies whether the media groups and default images of the product are returned (`false`) or not (`true`). If unspecified, defaults to `false`.

**Return Value**
Type: `ConnectApi.ProductCategoryDetailCollection`

**getProductCategoryPath(webserviceId, productCategoryId)**
Get the product category path from the root category to the current category.

**API Version**
49.0

**Available to Guest Users**
51.0

**Requires Chatter**
No
Signature

```java
public static ConnectApi.ProductCategoryPath getProductCategoryPath(String webstoreId, String productCategoryId)
```

Parameters

- **webstoreId**
  - Type: String
  - ID of the webstore.

- **productCategoryId**
  - Type: String
  - ID of the product category.

Return Value

Type: `ConnectApi.ProductCategoryPath`

Usage

This method respects buyer View Product entitlements and only users entitled to view product data can access it.

```java
getProductChildCollection(webstoreId, productId, effectiveAccountId, fields, mediaGroups, excludeFields, excludeMedia, excludeAttributeSetInfo, excludeQuantityRule, pageToken, pageSize)
```

Get a collection of child products related to a parent product.

API Version

57.0

Available to Guest Users

57.0

Requires Chatter

No

Signature

```java
public static ConnectApi.ProductChildCollection getProductChildCollection(String webstoreId, String productId, String effectiveAccountId, List<String> fields, List<String> mediaGroups, Boolean excludeFields, Boolean excludeMedia, Boolean excludeAttributeSetInfo, Boolean excludeQuantityRule, String pageToken, Integer pageSize)
```
Parameters

webstoreId
Type: String
ID of the webstore.

productId
Type: String
ID of the product.

effectiveAccountId
Type: String
ID of the buyer account or guest buyer profile for which the request is made. If unspecified, the default value is determined from context.

fields
Type: List<String>
Comma-separated list of field names.
If this list is empty or unspecified, all fields are returned. There is no limit to the number of fields you can specify. The number of fields and number of characters in the field name may affect the URL size limit. If excludeFields and fields are specified, the excludeFields parameter takes precedence.

mediaGroups
Type: List<String>
Comma-separated list of developer names of media group records.
If this list is empty or unspecified, all media groups are returned. If excludeMedia and mediaGroups are specified, the excludeMedia parameter takes precedence.

excludeFields
Type: Boolean
Specifies whether the fields are returned (false) or not (true). If unspecified, defaults to false.

excludeMedia
Type: Boolean
Specifies whether the media groups and default images of the product are returned (false) or not (true). If unspecified, defaults to false.

excludeAttributeSetInfo
Type: Boolean
Specifies whether the attribute set information for the product is returned (false) or not (true). If unspecified, defaults to false.

excludeQuantityRule
Type: Boolean
Specifies whether the quantity rule information for the product is returned (false) or not (true). If unspecified, defaults to false.

pageToken
Type: String
Specifies the base64 encoded page token. Page tokens are returned as part of the response. If unspecified, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you don't specify a value, the default size is 20.
Return Value
Type: `ConnectApi.ProductChildCollection`

`getProducts(webstoreId, effectiveAccountId, ids, skus, fields, excludeMedia, excludePrices)`
Get fields, prices, and default images for a list of products.

API Version
54.0

Available to Guest Users
54.0

Requires Chatter
No

Signature
`public static ConnectApi.ProductOverviewCollection getProducts(String webstoreId, String effectiveAccountId, List<String> ids, List<String> skus, List<String> fields, Boolean excludeMedia, Boolean excludePrices)`

Parameters

`webstoreId`
Type: `String`
ID of the webstore.

`effectiveAccountId`
Type: `String`
ID of the buyer account or guest buyer profile for which the request is made. If unspecified, the default value is determined from context.

`ids`
Type: `List<String>`
List of product IDs. Specify either a list of up to 20 product IDs or SKUs, but not both.

`skus`
Type: `List<String>`
List of SKUs. Specify either a list of up to 20 SKUs or product IDs, but not both.

`fields`
Type: `List<String>`
Comma-separated list of field names to return for each product. If the list is empty or not specified, all fields are returned. You can specify any number of fields.
excludeMedia
Type: Boolean
Specifies whether default images are returned for the products (false) or not (true). The default is false.

excludePrices
Type: Boolean
Specifies whether prices are returned for the products (false) or not (true). The default is false.

Note: In version 58.0 and later, prices aren’t returned for products regardless of this parameter. To get pricing information for products in version 58.0 and later, use the CommerceStorePricing Class.

Return Value
Type: ConnectApi.ProductOverviewCollection

getProducts(webstoreId, effectiveAccountId, ids, skus, fields, excludeMedia)
Get fields and default images for a list of products.

API Version
58.0

Available to Guest Users
58.0

Requires Chatter
No

Signature
public static ConnectApi.ProductOverviewCollection getProducts(String warehouseId, String effectiveAccountId, List<String> ids, List<String> skus, List<String> fields, Boolean excludeMedia)

Parameters
webstoreId
Type: String
ID of the webstore.
effectiveAccountId
Type: String
ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.
ids
Type: List<String>
List of product IDs. Specify either a list of up to 20 product IDs or SKUs, but not both.
skus
  Type: List<String>
  List of SKUs. Specify either a list of up to 20 SKUs or product IDs, but not both.

fields
  Type: List<String>
  Comma-separated list of field names to return for each product. If the list is empty or not specified, all fields are returned. You can specify any number of fields.

excludeMedia
  Type: Boolean
  Specifies whether default images are returned. Specifies whether the media groups and default images of the product are returned (false) or not (true). If unspecified, defaults to false.

Return Value
  Type: ConnectApi.ProductOverviewCollection

CommercePromotions Class
Evaluate promotions for Commerce orders. Get coupon code redemption usage.

Namespace
  ConnectApi

CommercePromotions Methods
The following are methods for CommercePromotions. All methods are static.

IN THIS SECTION:
  decreaseRedemption(couponCodeRedemption)
  Get coupon code redemption usage to revert a previously redeemed coupon code.
  evaluate(salesTransaction)
  Determine which promotions the customer is eligible for based on the store and buyer group, and compute the applicable price adjustments based on the coupons and the items in the cart. This API evaluates only the first 50 active manual promotions and first 50 active automatic promotions, based on priority. This API computes and returns applicable price adjustments, but it does not apply those adjustments to the webcart record. If you want to enable promotions based on shipping, contact Salesforce Customer Support.
  increaseRedemption(couponCodeRedemption)
  Get coupon code redemption addition (increase) usage.

**decreaseRedemption**(couponCodeRedemption)
Get coupon code redemption usage to revert a previously redeemed coupon code.
public static ConnectApi.CouponCodeRedemptionCollection
decreaseRedemption(ConnectApi.CouponCodeRedemptionInput couponCodeRedemption)

couponCodeRedemption
Type: ConnectApi.CouponCodeRedemptionInput on page 1579
Tracks each coupon code redemption.

Type: ConnectApi.CouponCodeRedemptionCollection on page 1767

evaluate(salesTransaction)
Determine which promotions the customer is eligible for based on the store and buyer group, and compute the applicable price adjustments based on the coupons and the items in the cart. This API evaluates only the first 50 active manual promotions and first 50 active automatic promotions, based on priority. This API computes and returns applicable price adjustments, but it does not apply those adjustments to the webcart record. If you want to enable promotions based on shipping, contact Salesforce Customer Support.

global static ConnectApi.PromotionEvaluation evaluate (ConnectApi.PromotionEvaluateInput salesTransaction)
Parameters

salesTransaction
Type: ConnectApi.PromotionEvaluateInput
Find promotions that the customer is eligible for and compute their discounts.

Return Value
Type: ConnectApi.PromotionEvaluation

increaseRedemption(couponCodeRedemption)
Get coupon code redemption addition (increase) usage.

API Version
58.0

Available to Guest Users
58.0

Requires Chatter
No

Signature

public static ConnectApi.CouponCodeRedemptionCollection
increaseRedemption(ConnectApi.CouponCodeRedemptionInput couponCodeRedemption)

Parameters

couponCodeRedemption
Type: ConnectApi.CouponCodeRedemptionInput on page 1579
Tracks each coupon code redemption.

Return Value
Type: ConnectApi.CouponCodeRedemptionCollection on page 1767

CommerceSearch Class

Namespace

ConnectApi
CommerceSearch Methods

These methods are for CommerceSearch. All methods are static.

IN THIS SECTION:

getSortRules(webstoreId)
Get sort rules for the live index.

getSuggestions(webstoreId, effectiveAccountId, searchTerm, maxResults)
Get suggestions for product searches.

getSuggestions(webstoreId, effectiveAccountId, searchTerm, maxResults, includeSuggestedProducts, maxSuggestedProducts)
Get suggestions for product searches.

searchProducts(webstoreId, effectiveAccountId, productSearchInput)
Search products.

getSortRules (webstoreId)
Get sort rules for the live index.

API Version
52.0

Available to Guest Users
52.0

Requires Chatter
No

Signature

class ConnectApi.SortRulesCollection
getSortRules(String webstoreId)

Parameters

webstoreId
Type: String
ID of the webstore.

Return Value
Type: ConnectApi.SortRulesCollection

getSuggestions (webstoreId, effectiveAccountId, searchTerm, maxResults)
Get suggestions for product searches.
API Version
52.0

Available to Guest Users
52.0

Requires Chatter
No

Signature
public static ConnectApi.ProductSearchSuggestionsResults getSuggestions(String webstoreId, String effectiveAccountId, String searchTerm, Integer maxResults)

Parameters
webstoreId
Type: String
ID of the webstore.
effectiveAccountId
Type: String
ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.
searchTerm
Type: String
Search term with up to 10 characters. If specified, the service returns autocomplete suggestions from the user’s recent searches. If unspecified, the service returns suggestions from the user’s recent searches.
maxResults
Type: Integer
Maximum number of suggestions. Values are from 1 through 10. If unspecified, defaults to 10.

Return Value
Type: ConnectApi.ProductSearchSuggestionsResults

getSuggestions(webstoreId, effectiveAccountId, searchTerm, maxResults, includeSuggestedProducts, maxSuggestedProducts)
Get suggestions for product searches.

API Version
58.0
Available to Guest Users

$8.0

Requires Chatter

No

Signature

public static ConnectApi.ProductSearchSuggestionsResults getSuggestions(String webstoreId, String effectiveAccountId, String searchTerm, Integer maxResults, Boolean includeSuggestedProducts, Integer maxSuggestedProducts)

Parameters

webstoreId
Type: String
ID of the webstore.

effectiveAccountId
Type: String
ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.

searchTerm
Type: String
Search term with up to 10 characters. If specified, the service returns autocomplete suggestions from the user’s recent searches. If unspecified, the service returns suggestions from the user’s recent searches.

maxResults
Type: Integer
Maximum number of suggestions. Values are from 1 through 10. If unspecified, defaults to 10.

includeSuggestedProducts
Type: Boolean
Specifies whether a search term returns product suggestions (true) or not (false). If unspecified, defaults to false. If true, the service returns the suggested product IDs.

maxSuggestedProducts
Type: String
Maximum number of product suggestions. Values are from 1 through 10. If unspecified, defaults to 4.

Return Value

Type: ConnectApi.ProductSearchSuggestionsResults

searchProducts(webstoreId, effectiveAccountId, productSearchInput)
Search products.
API Version
52.0

Available to Guest Users
52.0

Requires Chatter
No

Signature
public static ConnectApi.ProductSearchResults searchProducts(String webstoreId, String effectiveAccountId, ConnectApi.ProductSearchInput productSearchInput)

Parameters
webstoreId
Type: String
ID of the webstore.

effectiveAccountId
Type: String
ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.

productSearchInput
Type: ConnectApi.ProductSearchInput
A ConnectApi.ProductSearchInput body with either a category ID or search terms.

Return Value
Type: ConnectApi.ProductSearchResults

Usage
Searching products respects buyer View Product entitlements and only users entitled to view product data can access this resource.

CommerceSearchConnectFamily Class
Search products by search term or category in a webstore.

Namespace
ConnectApi

CommerceSearchConnectFamily Methods
The following are methods for CommerceSearchConnectFamily. All methods are static.
IN THIS SECTION:

```java
searchProducts(webstoreId, searchTerm, categoryId, sortRuleId, grouping, fields, refinements, pageParam, pageSize, effectiveAccountId, includeQuantityRule)
```

Search products by search term or category in a webstore.

**API Version**

58.0

**Available to Guest Users**

58.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.CommerceProductSearchResults searchProducts(String webstoreId, String searchTerm, String categoryId, String sortRuleId, String grouping, List<String> fields, String refinements, Integer pageParam, Integer pageSize, String effectiveAccountId, Boolean includeQuantityRule)
```

**Parameters**

- `webstoreId`
  
  Type: `String`
  
  ID of the webstore.

- `searchTerm`
  
  Type: `String`
  
  List of up to 32 space-separated search terms.

- `categoryId`
  
  Type: `String`
  
  Category ID returns results for products in this category or its subcategories.

- `sortRuleId`
  
  Type: `String`
  
  ID of the sort rule that specifies the order of products in the search results. If unspecified, the default sort type is relevancy.

- `grouping`
  
  Type: `String`
Grouping option for search results. If unspecified, the default is the value specified in **Search > Results Display Settings > Results Grouping**.

**fields**
Type: `List<String>`
Product fields to return in search results. Search results include fields you have access to.

**refinements**
Type: `String`
List up to nine refinements (facets) for search results. Buyers or shoppers can select up to 20 values for each refinement. The `refinements` parameter is encoded as a Base64 string from the JSON representation of `ConnectApi.DistinctValueRefinementInput`.

**pageParam**
Type: `Integer`
Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

**pageSize**
Type: `Integer`
Specifies the number of items per page. Valid values are from 1 through 200. If unspecified, defaults to 20.

**effectiveAccountId**
Type: `String`
ID of the buyer account or guest buyer profile for which the request is made. If unspecified, the default value is determined from context.

**includeQuantityRule**
Type: `Boolean`
Specifies whether to include purchase quantity rule information for products in search results (`true`) or not (`false`). If unspecified, defaults to `false`.

**Return Value**
Type: `ConnectApi.CommerceProductSearchResults`

**Usage**
Searching products respects buyer View Product entitlements and only users entitled to view product data can access this resource.

**CommerceSearchSettings Class**
Get indexes. Get index logs. Create an index of a product catalog.

**Namespace**
`ConnectApi`

**CommerceSearchSettings Methods**
These methods are for `CommerceSearchSettings`. All methods are static.
IN THIS SECTION:

createCommerceSearchIndex(webstoreId, indexBuildType)
Create an index of a product catalog.

getCommerceSearchIndex(webstoreId, indexId)
Get a Commerce search index.

getCommerceSearchIndexes(webstoreId)
Get Commerce search indexes.

getCommerceSearchIndexLogs(webstoreId)
Get Commerce search index logs.

createCommerceSearchIndex(webstoreId, indexBuildType)
Create an index of a product catalog.

API Version
57.0

Requires Chatter
Yes

Signature
public static ConnectApi.CommerceSearchIndex postCommerceSearchIndex(String webstoreId)

Parameters
webstoreId
Type: String
ID of the webstore.

indexBuildType
Type: ConnectApi.CommerceSearchIndexBuildType
Build type of the index. Values are:

• Full
• Incremental

Return Value
Type: ConnectApi.CommerceSearchIndex

Usage
This method creates a live index that replaces the current live index. Any indexes that are in progress are removed when you manually create an index with this method.
getCommerceSearchIndex(webstoreId, indexId)
Get a Commerce search index.

API Version
52.0

Requires Chatter
Yes

Signature
public static ConnectApi.CommerceSearchIndex getSingleCommerceSearchIndex(String webstoreId, String indexId)

Parameters
webstoreId
Type: String
ID of the webstore.

indexId
Type: String
ID of the index.

Return Value
Type: ConnectApi.CommerceSearchIndex

getCommerceSearchIndexes(webstoreId)
Get Commerce search indexes.

API Version
52.0

Requires Chatter
Yes

Signature
public static ConnectApi.CommerceSearchIndexCollection getCommerceSearchIndexes(String webstoreId)
Parameters

webstoreId
Type: String
ID of the webstore.

Return Value

Type: ConnectApi.CommerceSearchIndexCollection

getCommerceSearchIndexLogs (webstoreId)
Get Commerce search index logs.

API Version

57.0

Requires Chatter

No

Signature

public static ConnectApi.CommerceSearchIndexLogCollection getCommerceSearchIndexLogs(String webstoreId)

Parameters

webstoreId
Type: String
ID of the webstore.

Return Value

Type: ConnectApi.CommerceSearchIndexLogCollection

CommerceStorePricing Class

Get product prices.

Namespace

ConnectApi

CommerceStorePricing Methods

These methods are for CommerceStorePricing. All methods are static.
IN THIS SECTION:

- getProductPrice(webstoreId, productId, effectiveAccountId)
  Get the list and buyer price for a product.
- getProductPrice(webstoreId, productId, effectiveAccountId, productSellingModelIds)
  Get a product's list and buyer price for specified product selling models.
- getProductPrices(webstoreId, effectiveAccountId, pricingInput)
  Get the prices for multiple products.
- getProductPrices(webstoreId, effectiveAccountId, productIds)
  Get the prices for multiple products.

**getProductPrice(webstoreId, productId, effectiveAccountId)**

Get the list and buyer price for a product.

**API Version**

49.0

**Available to Guest Users**

51.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.ProductPrice getProductPrice(String webstoreId, String productId, String effectiveAccountId)
```

**Parameters**

- `webstoreId`
  Type: String
  ID of the webstore.
- `productId`
  Type: String
  ID of the product.
- `effectiveAccountId`
  Type: String
  ID of the buyer account or guest buyer profile for which the request is made.

**Return Value**

Type: `ConnectApi.ProductPrice`
Usage

This method respects buyer entitlements and only users entitled to view product and price data can access it.

If a store is segmented into markets, this API looks at the language parameter appended to the URL to determine the shopper’s locale and returns the appropriate values.

```
getProductPrice(webstoreId, productId, effectiveAccountId, productSellingModelIds)
```

Get a product’s list and buyer price for specified product selling models.

API Version

56.0

Available to Guest Users

56.0

Requires Chatter

No

Signature

```
public static ConnectApi.ProductPrice getProductPrice(String webstoreId, String productId, String effectiveAccountId, List<String> productSellingModelIds)
```

Parameters

`webstoreId`  
Type: String  
ID of the webstore.

`productId`  
Type: String  
ID of the product.

`effectiveAccountId`  
Type: String  
ID of the buyer account or guest buyer profile for which the request is made.

`productSellingModelIds`  
Type: List<String>  
List of product selling model IDs for the product.

Return Value

Type: ConnectApi.ProductPrice
getProductPrices(webstoreId, effectiveAccountId, pricingInput)
Get the prices for multiple products.

API Version
49.0

Available to Guest Users
51.0

Requires Chatter
No

Signature
public static ConnectApi.PricingResult getProductPrices(String webstoreId, String effectiveAccountId, ConnectApi.PricingInput pricingInput)

Parameters
webstoreId
  Type: String
  ID of the webstore.
n
effectiveAccountId
  Type: String
  ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.
n
pricingInput
  Type: ConnectApi.PricingInput
  A ConnectApi.PricingInput body with the list of line items to price.

Return Value
Type: ConnectApi.PricingResult

Usage
This method respects buyer entitlements and only users entitled to view product and price data can access it.
If a store is segmented into markets, this API looks at the language parameter appended to the URL to determine the shopper’s locale and returns the appropriate values.

getProductPrices(webstoreId, effectiveAccountId, productIds)
Get the prices for multiple products.
API Version
54.0

Available to Guest Users
54.0

Requires Chatter
No

Signature
public static ConnectApi.PricingResult getProductPrices(String webstoreId, String effectiveAccountId, List<String> productIds)

Parameters
webstoreId
Type: String
ID of the webstore.
effectiveAccountId
Type: String
ID of the buyer account or guest buyer profile for which the request is made. If null, the default value is determined from context.
productIds
Type: List<String>
List of product IDs for which you want to get prices.

Return Value
Type: ConnectApi.PricingResult

Usage
This method respects buyer entitlements and only users entitled to view product and price data can access it.
If a store is segmented into markets, this API looks at the language parameter appended to the URL to determine the shopper’s locale and returns the appropriate values.

CommerceWishlist Class
Get, create, update, and delete wishlists. Add wishlists to carts. Get wishlist items, add items to wishlists, and delete wishlist items.

Namespace
ConnectApi
CommerceWishlist Methods

These methods are for CommerceWishlist. All methods are static.

IN THIS SECTION:

- `addItemToWishlist(webstoreId, wishlistId, wishlistItemInput)`
  Add an item to a wishlist for the context user.

- `addItemToWishlist(webstoreId, effectiveAccountId, wishlistId, wishlistItemInput)`
  Add an item to a wishlist.

- `addWishlistToCart(webstoreId, wishlistId)`
  Add a wishlist to the active cart for the context user.

- `addWishlistToCart(webstoreId, wishlistId, effectiveAccountId)`
  Add a wishlist to the active cart.

- `addWishlistToCartWithCartId(webstoreId, wishlistId, cartId)`
  Add a wishlist to a cart.

- `addWishlistToCartWithCartId(webstoreId, wishlistId, cartId, effectiveAccountId)`
  Add a wishlist to a cart.

- `createWishlist(webstoreId, wishlistInput)`
  Create a wishlist for the context user.

- `createWishlist(webstoreId, effectiveAccountId, wishlistInput)`
  Create a wishlist.

- `deleteWishlist(webstoreId, wishlistId)`
  Delete a wishlist for the context user.

- `deleteWishlist(webstoreId, effectiveAccountId, wishlistId)`
  Delete a wishlist.

- `getWishlist(webstoreId, effectiveAccountId, wishlistId, productFields, sortItemsBy)`
  Get a wishlist with product fields sorted by items.

- `getWishlist(webstoreId, effectiveAccountId, wishlistId, productFields, pageSize, sortItemsBy)`
  Get a wishlist with product fields sorted by items with a specified number of items per page.

- `getWishlistItems(webstoreId, effectiveAccountId, wishlistId, productFields, pageParam, sortItemsBy)`
  Get a page of sorted wishlist items with product fields.

- `getWishlistItems(webstoreId, effectiveAccountId, wishlistId, productFields, pageParam, pageSize, sortItemsBy)`
  Get a page of specified size of sorted wishlist items with product fields.

- `getWishlistSummaries(webstoreId, effectiveAccountId, includeDisplayedList)`
  Get wishlist summaries.

- `getWishlistSummaries(webstoreId, effectiveAccountId, includeDisplayedList, productFields, sortItemsBy)`
  Get wishlist summaries with product fields sorted by items.

- `getWishlistSummaries(webstoreId, effectiveAccountId, includeDisplayedList, productFields, pageSize, sortItemsBy)`
  Get wishlist summaries with product fields sorted by items with a specified number of items per page.

- `removeWishlistItem(webstoreId, effectiveAccountId, wishlistId, wishlistItemId)`
  Remove an item from a wishlist.
updateWishlist(webstoreId, wishlistId, wishlistUpdateInput)
Update the name of a wishlist for the context user.

updateWishlist(webstoreId, effectiveAccountId, wishlistId, wishlistUpdateInput)
Update the name of a wishlist.

addItemToWishlist(webstoreId, wishlistId, wishlistItemInput)
Add an item to a wishlist for the context user.

API Version
49.0

Requires Chatter
No

Signature
public static ConnectApi.WishlistItem addItemToWishlist(String webstoreId, String wishlistId, ConnectApi.WishlistItemInput wishlistItemInput)

Parameters
webstoreId
Type: String
ID of the webstore.

wishlistId
Type: String
ID of the wishlist.

wishlistItemInput
Type: ConnectApi.WishlistItemInput
A ConnectApi.WishlistItemInput body with the item to add to the wishlist.

Return Value
Type: ConnectApi.WishlistItem

Usage
This method respects buyer View Product entitlements and only users entitled to view product data can access it.

addItemToWishlist(webstoreId, effectiveAccountId, wishlistId, wishlistItemInput)
Add an item to a wishlist.
API Version
49.0

Requires Chatter
No

Signature
public static ConnectApi.WishlistItem addItemToWishlist(String webstoreId, String effectiveAccountId, String wishlistId, ConnectApi.WishlistItemInput wishlistItemInput)

Parameters
webstoreId
Type: String
ID of the webstore.
effectiveAccountId
Type: String
ID of the account for which the request is made. If null, defaults to the account ID for the context user.
wishlistId
Type: String
ID of the wishlist.
wishlistItemInput
Type: ConnectApi.WishlistItemInput
A ConnectApi.WishlistItemInput body with the item to add to the wishlist.

Return Value
Type: ConnectApi.WishlistItem

Usage
This method respects buyer View Product entitlements and only users entitled to view product data can access it.

addWishlistToCart(webstoreId, wishlistId)
Add a wishlist to the active cart for the context user.

API Version
49.0

Requires Chatter
No
Signature

```java
public static ConnectApi.WishlistToCartResult addWishlistToCart(String webstoreId, String wishlistId)
```

Parameters

- **webstoreId**
  - Type: String
  - ID of the webstore.
- **wishlistId**
  - Type: String
  - ID of the wishlist.

Return Value

Type: `ConnectApi.WishlistToCartResult`

Usage

This method respects buyer View Product entitlements and only users entitled to view product data can access it.

```java
addWishlistToCart(webstoreId, wishlistId, effectiveAccountId)
```

Add a wishlist to the active cart.

API Version

49.0

Requires Chatter

No

Signature

```java
public static ConnectApi.WishlistToCartResult addWishlistToCart(String webstoreId, String wishlistId, String effectiveAccountId)
```

Parameters

- **webstoreId**
  - Type: String
  - ID of the webstore.
- **wishlistId**
  - Type: String
  - ID of the wishlist.
- **effectiveAccountId**
  - Type: String
ID of the account for which the request is made. If null, defaults to the account ID for the context user.

**Return Value**
Type: `ConnectApi.WishlistToCartResult`

**Usage**
This method respects buyer View Product entitlements and only users entitled to view product data can access it.

```java
addWishlistToCartWithCartId(webstoreId, wishlistId, cartId)
```
Add a wishlist to a cart.

**API Version**
49.0

**Requires Chatter**
No

**Signature**
```
public static ConnectApi.WishlistToCartResult addWishlistToCartWithCartId(String webstoreId, String wishlistId, String cartId)
```

**Parameters**
- `webstoreId`
  Type: `String`
  ID of the webstore.
- `wishlistId`
  Type: `String`
  ID of the wishlist.
- `cartId`
  Type: `String`
  ID of the cart. If null, wishlist items are added to the active cart.

**Return Value**
Type: `ConnectApi.WishlistToCartResult`

**Usage**
This method respects buyer View Product entitlements and only users entitled to view product data can access it.
addWishlistToCartWithCartId(webstoreId, wishlistId, cartId, effectiveAccountId)

Add a wishlist to a cart.

API Version
49.0

Requires Chatter
No

Signature
public static ConnectApi.WishlistToCartResult addWishlistToCartWithCartId(String webstoreId, String wishlistId, String cartId, String effectiveAccountId)

Parameters
webstoreId
Type: String
ID of the webstore.

wishlistId
Type: String
ID of the wishlist.

cartId
Type: String
ID of the cart. If null, wishlist items are added to the active cart.

effectiveAccountId
Type: String
ID of the account for which the request is made. If null, defaults to the account ID for the context user.

Return Value
Type: ConnectApi.WishlistToCartResult

Usage
This method respects buyer View Product entitlements and only users entitled to view product data can access it.

createWishlist(webstoreId, wishlistInput)

Create a wishlist for the context user.

API Version
49.0
Requires Chatter
No

Signature
public static ConnectApi.Wishlist createWishlist(String webstoreId, ConnectApi.WishlistInput wishlistInput)

Parameters
webstoreId
  Type: String
  ID of the webstore.

wishlistInput
  Type: ConnectApi.WishlistInput
  A ConnectApi.WishlistInput body that includes the wishlist name and items.

Return Value
Type: ConnectApi.Wishlist

createWishlist(webstoreId, effectiveAccountId, wishlistInput)
Create a wishlist.

API Version
49.0

Requires Chatter
No

Signature
public static ConnectApi.Wishlist createWishlist(String webstoreId, String effectiveAccountId, ConnectApi.WishlistInput wishlistInput)

Parameters
webstoreId
  Type: String
  ID of the webstore.

effectiveAccountId
  Type: String
  ID of the account for which the request is made. If null, defaults to the account ID for the context user.

wishlistInput
  Type: ConnectApi.WishlistInput
A ConnectApi.WishlistInput body that includes the wishlist name and items.

Return Value
Type: ConnectApi.Wishlist

deleteWishlist(webstoreId, wishlistId)
Delete a wishlist for the context user.

API Version
49.0

Requires Chatter
No

Signature
public static Void deleteWishlist(String webstoreId, String wishlistId)

Parameters
webstoreId
Type: String
ID of the webstore.
wishlistId
Type: String
ID of the wishlist.

Return Value
Type: Void

deleteWishlist(webstoreId, effectiveAccountId, wishlistId)
Delete a wishlist.

API Version
51.0

Requires Chatter
No
Signature

public static Void deleteWishlist(String webstoreId, String effectiveAccountId, String wishlistId)

Parameters

webstoreId
  Type: String
  ID of the webstore.

effectiveAccountId
  Type: String
  ID of the account for which the request is made. If null, defaults to the account ID for the context user.

wishlistId
  Type: String
  ID of the wishlist.

Return Value

Type: Void

gEWishlist(webstoreId, effectiveAccountId, wishlistId, productFields, sortItemsBy)

Get a wishlist with product fields sorted by items.

API Version

S1.0

Requires Chatter

No

Signature

public static ConnectApi.Wishlist getWishlist(String webstoreId, String effectiveAccountId, String wishlistId, String productFields, ConnectApi.WishlistItemSortOrder sortItemsBy)

Parameters

webstoreId
  Type: String
  ID of the webstore.

effectiveAccountId
  Type: String
  ID of the account for which the request is made. If null, defaults to the account ID for the context user.
wishlistId
   Type: String
   ID of the wishlist.

productFields
   Type: String
   Comma-separated list of custom product fields. Fields aren’t case-sensitive. For example, ProductCode and productcode return the same results.

sortItemsBy
   Type: ConnectApi.WishlistItemSortOrder
   Sort order for wishlist items. Values are:
   • CreatedDateAsc—Sorts by oldest creation date.
   • CreatedDateDesc—Sorts by most recent creation date.
   The default sort order is CreatedDateDesc.

Return Value
Type: ConnectApi.Wishlist

getWishlist(webstoreId, effectiveAccountId, wishlistId, productFields, pageSize, sortItemsBy)
Get a wishlist with product fields sorted by items with a specified number of items per page.

API Version
S1.0

Requires Chatter
No

Signature
public static ConnectApi.Wishlist getWishlist(String webstoreId, String effectiveAccountId, String wishlistId, String productFields, Integer pageSize, ConnectApi.WishlistItemSortOrder sortItemsBy)

Parameters
webstoreId
   Type: String
   ID of the webstore.

effectiveAccountId
   Type: String
   ID of the account for which the request is made. If null, defaults to the account ID for the context user.
wishlistId
  Type: String
  ID of the wishlist.

productFields
  Type: String
  Comma-separated list of custom product fields. Fields aren’t case-sensitive. For example, ProductCode and productcode return the same results.

pageSize
  Type: Integer
  Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortItemsBy
  Type: ConnectApi.WishlistItemSortOrder
  Sort order for wishlist items. Values are:
  • CreatedDateAsc—Sorts by oldest creation date.
  • CreatedDateDesc—Sorts by most recent creation date.
  The default sort order is CreatedDateDesc.

Return Value
  Type: ConnectApi.Wishlist

getWishlistItems(webstoreId, effectiveAccountId, wishlistId, productFields, pageParam, sortItemsBy)
Get a page of sorted wishlist items with product fields.

API Version
51.0

Requires Chatter
No

Signature
public static ConnectApi.WishlistItemCollection getWishlistItems(String webstoreId, String effectiveAccountId, String wishlistId, String productFields, String pageParam, ConnectApi.WishlistItemSortOrder sortItemsBy)

Parameters
webstoreId
  Type: String
  ID of the webstore.
**effectiveAccountId**
Type: **String**
ID of the account for which the request is made. If **null**, defaults to the account ID for the context user.

**wishlistId**
Type: **String**
ID of the wishlist.

**productFields**
Type: **String**
Comma-separated list of custom product fields. Fields aren’t case-sensitive. For example, ProductCode and productcode return the same results.

**pageParam**
Type: **String**
Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in **null**, the first page is returned.

**sortItemsBy**
Type: **ConnectApi.WishlistItemSortOrder**
Sort order for wishlist items. Values are:

- **CreatedDateAsc**—Sorts by oldest creation date.
- **CreatedDateDesc**—Sorts by most recent creation date.

The default sort order is **CreatedDateDesc**.

**Return Value**
Type: **ConnectApi.WishlistItemCollection**

**getWishlistItems** *(webstoreId, effectiveAccountId, wishlistId, productFields, pageParam, pageSize, sortItemsBy)*
Get a page of specified size of sorted wishlist items with product fields.

**API Version**
51.0

**Requires Chatter**
No

**Signature**
```
public static ConnectApi.WishlistItemCollection getWishlistItems(String webstoreId, String effectiveAccountId, String wishlistId, String productFields, String pageParam, Integer pageSize, ConnectApi.WishlistItemSortOrder sortItemsBy)
```
Parameters

webstoreId
  Type: String
  ID of the webstore.

effectiveAccountId
  Type: String
  ID of the account for which the request is made. If null, defaults to the account ID for the context user.

wishlistId
  Type: String
  ID of the wishlist.

productFields
  Type: String
  Comma-separated list of custom product fields. Fields aren’t case-sensitive. For example, ProductCode and productcode return the same results.

pageParam
  Type: String
  Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
  Type: Integer
  Specifies the number of items per page. Valid values are from 1 through 200. If you pass in null, the default size is 25.

sortItemsBy
  Type: ConnectApi.WishlistItemSortOrder
  Sort order for wishlist items. Values are:
  • CreatedDateAsc—Sorts by oldest creation date.
  • CreatedDateDesc—Sorts by most recent creation date.
  The default sort order is CreatedDateDesc.

Return Value

Type: ConnectApi.WishlistItemCollection

getWishlistSummaries(webstoreId, effectiveAccountId, includeDisplayedList)

Get wishlist summaries.

API Version

49.0

Requires Chatter

No
Signature

```java
public static ConnectApi.WishlistsSummary getWishlistSummaries(String webstoreId, String effectiveAccountId, Boolean includeDisplayedList)
```

Parameters

- **webstoreId**
  - **Type:** String
  - ID of the webstore.

- **effectiveAccountId**
  - **Type:** String
  - ID of the account for which the request is made. If `null`, defaults to the account ID for the context user.

- **includeDisplayedList**
  - **Type:** Boolean
  - Specifies whether to include the displayed list (`true`) or not (`false`). If `null`, defaults to `false`.

Return Value

- **Type:** `ConnectApi.WishlistsSummary`

```java
getWishlistSummaries(webstoreId, effectiveAccountId, includeDisplayedList, productFields, sortItemsBy)
```

Get wishlist summaries with product fields sorted by items.

API Version

- **51.0**

Requires Chatter

- **No**

Signature

```java
public static ConnectApi.WishlistsSummary getWishlistSummaries(String webstoreId, String effectiveAccountId, Boolean includeDisplayedList, String productFields, ConnectApi.WishlistItemSortOrder sortItemsBy)
```

Parameters

- **webstoreId**
  - **Type:** String
  - ID of the webstore.

- **effectiveAccountId**
  - **Type:** String
  - ID of the account for which the request is made. If `null`, defaults to the account ID for the context user.
**includeDisplayedList**
Type: Boolean
Specifies whether to include the displayed list (true) or not (false).

**productFields**
Type: String
Comma-separated list of custom product fields. Fields aren’t case-sensitive. For example, ProductCode and productcode return the same results.

If includeDisplayedList is false, productFields is ignored.

**sortItemsBy**
Type: ConnectApi.WishlistItemSortOrder
Sort order for wishlist items. Values are:
- CreatedDateAsc—Sorts by oldest creation date.
- CreatedDateDesc—Sorts by most recent creation date.

The default sort order is CreatedDateDesc.
If includeDisplayedList is false, sortItemsBy is ignored.

**Return Value**
Type: ConnectApi.WishlistsSummary

**getWishlistSummaries(webstoreId, effectiveAccountId, includeDisplayedList, productFields, pageSize, sortItemsBy)**
Get wishlist summaries with product fields sorted by items with a specified number of items per page.

**API Version**
51.0

**Requires Chatter**
No

**Signature**
public static ConnectApi.WishlistsSummary getWishlistSummaries(String webstoreId, String effectiveAccountId, Boolean includeDisplayedList, Integer pageSize, String productFields, pageSize, ConnectApi.WishlistItemSortOrder sortItemsBy)

**Parameters**
webstoreId
Type: String
ID of the webstore.
effectiveAccountId
    Type: String
    ID of the account for which the request is made. If null, defaults to the account ID for the context user.

includeDisplayedList
    Type: Boolean
    Specifies whether to include the displayed list (true) or not (false).

productFields
    Type: String
    Comma-separated list of custom product fields. Fields aren’t case-sensitive. For example, ProductCode and productcode return the same results.
    If includeDisplayedList is false, productFields is ignored.

pageSize
    Type: Integer
    Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortItemsBy
    Type: ConnectApi.WishlistItemSortOrder
    Sort order for wishlist items. Values are:
    • CreatedDateAsc—Sorts by oldest creation date.
    • CreatedDateDesc—Sorts by most recent creation date.
    The default sort order is CreatedDateDesc.
    If includeDisplayedList is false, sortItemsBy is ignored.

Return Value
    Type: ConnectApi.WishlistsSummary

removeWishlistItem(webstoreId, effectiveAccountId, wishlistId, wishlistItemId)
Remove an item from a wishlist.

API Version
49.0

Requires Chatter
No

Signature
public static Void removeWishlistItem(String webstoreId, String effectiveAccountId, String wishlistId, String wishlistItemId)
Parameters

webstoreId  
Type: String  
ID of the webstore.

effectiveAccountId  
Type: String  
ID of the account for which the request is made. If null, defaults to the account ID for the context user.

wishlistId  
Type: String  
ID of the wishlist.

wishlistItemId  
Type: String  
ID of the wishlist item to remove.

Return Value

Type: Void

updateWishlist(webstoreId, wishlistId, wishlistUpdateInput)

Update the name of a wishlist for the context user.

API Version

50.0

Requires Chatter

No

Signature

public static ConnectApi.WishlistSummary updateWishlist(String webstoreId, String wishlistId, ConnectApi.WishlistUpdateInput wishlistUpdateInput)

Parameters

webstoreId  
Type: String  
ID of the webstore.

wishlistId  
Type: String  
ID of the wishlist.

wishlistUpdateInput  
Type: ConnectApi.WishlistUpdateInput  
A ConnectApi.WishlistUpdateInput body with the wishlist name to update.
Return Value
Type: ConnectApi.WishlistSummary

updateWishlist(webstoreId, effectiveAccountId, wishlistId, wishlistUpdateInput)
Update the name of a wishlist.

API Version
50.0

Requires Chatter
No

Signature
public static ConnectApi.WishlistSummary updateWishlist(String webstoreId, String effectiveAccountId, String wishlistId, ConnectApi.WishlistUpdateInput wishlistUpdateInput)

Parameters
webstoreId
Type: String
ID of the webstore.
effectiveAccountId
Type: String
ID of the account for which the request is made. If null, defaults to the account ID for the context user.
wishlistId
Type: String
ID of the wishlist.
wishlistUpdateInput
Type: ConnectApi.WishlistUpdateInput
A ConnectApi.WishlistUpdateInput body with the wishlist name to update.

Return Value
Type: ConnectApi.WishlistSummary

Communities Class
Get information about Experience Cloud sites in your org.
Namespace
ConnectApi

Communities Methods
The following are methods for Communities. All methods are static.

IN THIS SECTION:
getCommunities()
Get a list of Experience Cloud sites that the context user has access to.
getCommunities(communityStatus)
Get a list of Experience Cloud sites with the specified status that the context user has access to.
getCommunity(communityId)
Get information about an Experience Cloud site.

getCommunities()
Get a list of Experience Cloud sites that the context user has access to.

API Version
28.0

Requires Chatter
No

Signature
public static ConnectApi.CommunityPage getCommunities()

Return Value
Type: ConnectApi.CommunityPage

getCommunities(communityStatus)
Get a list of Experience Cloud sites with the specified status that the context user has access to.

API Version
28.0

Requires Chatter
No
Signature

`public static ConnectApi.CommunityPage getCommunities(ConnectApi.CommunityStatus communityStatus)`

Parameters

`communityStatus`

Type: `ConnectApi.CommunityStatus`

`communityStatus`—Status of the Experience Cloud site. Values are:

- Live
- Inactive
- UnderConstruction

Return Value

Type: `ConnectApi.CommunityPage`

`getCommunity(String communityId)`

Get information about an Experience Cloud site.

API Version

28.0

Available to Guest Users

35.0

Requires Chatter

No

Signature

`public static ConnectApi.Community getCommunity(String communityId)`

Parameters

`communityId`

Type: `String`

ID of an Experience Cloud site. You can’t specify `null` or `internal`.

Return Value

Type: `ConnectApi.Community`
CommunityModeration Class

Get information about flagged feed items and comments in an Experience Cloud site. Add and remove flags from comments and feed items.

Namespace

ConnectApi

CommunityModeration Methods

The following are methods for CommunityModeration. All methods are static.

IN THIS SECTION:

addFlagToComment(communityId, commentId)
Add a moderation flag to a comment.

addFlagToComment(communityId, commentId, visibility)
Add a moderation flag of the specified visibility to a comment.

addFlagToComment(communityId, commentId, type)
Add a moderation flag of the specified type to a comment.

addFlagToComment(communityId, commentId, note)
Add a moderation flag with a note to a comment.

addFlagToComment(communityId, commentId, type, note)
Add a moderation flag of the specified type with a note to a comment.

addFlagToComment(communityId, commentId, type, visibility)
Add a moderation flag of the specified type and visibility to a comment.

addFlagToFeedElement(communityId, feedElementId)
Add a moderation flag to a feed element.

addFlagToFeedElement(communityId, feedElementId, visibility)
Add a moderation flag of the specified visibility to a feed element.

addFlagToFeedElement(communityId, feedElementId, type)
Add a moderation flag of the specified type to a feed element.

addFlagToFeedElement(communityId, feedElementId, note)
Add a moderation flag with a note to a feed element.

addFlagToFeedElement(communityId, feedElementId, type, note)
Add a moderation flag of the specified type with a note to a feed element.

addFlagToFeedElement(communityId, feedElementId, type, visibility)
Add a moderation flag of the specified type and visibility to a feed element.
addFlagToFeedElement(communityId, feedElementId, visibility, note)
Add a moderation flag of the specified visibility with a note to a feed element.

addFlagToFeedElement(communityId, feedElementId, type, visibility, note)
Add a moderation flag of the specified type and visibility with a note to a feed element.

getFlagsOnComment(communityId, commentId)
Get the moderation flags on a comment.

getFlagsOnComment(communityId, commentId, visibility)
Get the moderation flags with specified visibility on a comment.

getFlagsOnComment(communityId, commentId, pageSize, pageParam)
Get a page of moderation flags on a comment.

getFlagsOnComment(communityId, commentId, visibility, pageSize, pageParam)
Get a page of moderation flags with specified visibility on a comment.

getFlagsOnFeedElement(communityId, feedElementId)
Get the moderation flags on a feed element.

getFlagsOnFeedElement(communityId, feedElementId, visibility)
Get the moderation flags with specified visibility on a feed element.

getFlagsOnFeedElement(communityId, feedElementId, pageParam, pageSize)
Get a page of moderation flags on a feed element.

getFlagsOnFeedElement(communityId, feedElementId, visibility, pageParam, pageSize)
Get a page of moderation flags with specified visibility on a feed element.

removeFlagFromComment(communityId, commentId, userId)
Remove a moderation flag from a comment.

removeFlagFromFeedElement(communityId, feedElementId, userId)
Remove a moderation flag from a feed element.

addFlagToComment(communityId, commentId)
Add a moderation flag to a comment.

API Version
29.0

Requires Chatter
Yes

Signature
public static ConnectApi.ModerationFlags addFlagToComment(String communityId, String commentId)
Parameters

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or null.

- **commentId**
  - Type: `String`
  - ID for a comment.

Return Value

Type: `ConnectApi.ModerationFlags`

Usage

To add a flag to a comment, Allow members to flag content must be selected for an Experience Cloud site.

**addFlagToComment(communityId, commentId, visibility)**

Add a moderation flag of the specified visibility to a comment.

API Version

30.0

Requires Chatter

Yes

Signature

```
public static ConnectApi.ModerationFlags addFlagToComment(String communityId, String commentId, ConnectApi.CommunityFlagVisibility visibility)
```

Parameters

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or null.

- **commentId**
  - Type: `String`
  - ID for a comment.

- **visibility**
  - Type: `ConnectApi.CommunityFlagVisibility`
  - Visibility behavior of a flag for various user types.
  - **ModeratorsOnly**—The flag is visible only to users with moderation permissions on the flagged element or item.
• SelfAndModerators—The flag is visible to the creator of the flag and to users with moderation permissions on the flagged element or item.

Return Value
Type: ConnectApi.ModerationFlags

Usage
To add a flag to a comment, Allow members to flag content must be selected for an Experience Cloud site.

addFlagToComment(communityId, commentId, type)
Add a moderation flag of the specified type to a comment.

API Version
38.0

Requires Chatter
Yes

Signature
public static ConnectApi.ModerationFlags addFlagToComment(String communityId, String commentId, ConnectApi.CommunityFlagType type)

Parameters
- communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

- commentId
  Type: String
  ID for a comment.

- type
  Type: ConnectApi.CommunityFlagType
  Type of moderation flag.
  - FlagAsInappropriate—Flag for inappropriate content.
  - FlagAsSpam—Flag for spam.
  If a type isn’t specified, it defaults to FlagAsInappropriate.

Return Value
Type: ConnectApi.ModerationFlags
Usage
To add a flag to a comment, Allow members to flag content must be selected for an Experience Cloud site.

`addFlagToComment(communityId, commentId, note)`
Add a moderation flag with a note to a comment.

API Version
38.0

Requires Chatter
Yes

Signature
`public static ConnectApi.ModerationFlags addFlagToComment(String communityId, String commentId, String note)`

Parameters
- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.
- `commentId`
  Type: `String`
  ID for a comment.
- `note`
  Type: `String`
  A note of up to 4,000 characters about the flag.

Return Value
Type: `ConnectApi.ModerationFlags`

Usage
To add a flag to a comment, Allow members to flag content must be selected for an Experience Cloud site.

`addFlagToComment(communityId, commentId, type, note)`
Add a moderation flag of the specified type with a note to a comment.

API Version
38.0
Requires Chatter
Yes

Signature
public static ConnectApi.ModerationFlags addFlagToComment(String communityId, String commentId, ConnectApi.CommunityFlagType type, String note)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

commentId
Type: String
ID for a comment.

type
Type: ConnectApi.CommunityFlagType
Type of moderation flag.
- FlagAsInappropriate—Flag for inappropriate content.
- FlagAsSpam—Flag for spam.

If a type isn’t specified, it defaults to FlagAsInappropriate.

note
Type: String
A note of up to 4,000 characters about the flag.

Return Value
Type: ConnectApi.ModerationFlags

Usage
To add a flag to a comment, Allow members to flag content must be selected for an Experience Cloud site.

addFlagToComment(communityId, commentId, type, visibility)
Add a moderation flag of the specified type and visibility to a comment.

API Version
38.0

Requires Chatter
Yes
Signature

```java
public static ConnectApi.ModerationFlags addFlagToComment(String communityId, String commentId, ConnectApi.CommunityFlagType type, ConnectApi.CommunityFlagVisibility visibility)
```

Parameters

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **commentId**
  - Type: `String`
  - ID for a comment.

- **type**
  - Type: `ConnectApi.CommunityFlagType`
  - Type of moderation flag.
    - FlagAsInappropriate—Flag for inappropriate content.
    - FlagAsSpam—Flag for spam.

If a type isn't specified, it defaults to FlagAsInappropriate.

- **visibility**
  - Type: `ConnectApi.CommunityFlagVisibility`
  - Visibility behavior of a flag for various user types.
    - ModeratorsOnly—The flag is visible only to users with moderation permissions on the flagged element or item.
    - SelfAndModerators—The flag is visible to the creator of the flag and to users with moderation permissions on the flagged element or item.

Return Value

- Type: `ConnectApi.ModerationFlags`

Usage

To add a flag to a comment, Allow members to flag content must be selected for an Experience Cloud site.

```java
addFlagToComment(communityId, commentId, visibility, note)
```

Add a moderation flag of the specified visibility with a note to a comment.

API Version

- 38.0

Requires Chatter

- Yes
public static ConnectApi.ModerationFlags addFlagToComment(String communityId, String commentId, ConnectApi.CommunityFlagVisibility visibility, String note)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

commentId
Type: String
ID for a comment.

visibility
Type: ConnectApi.CommunityFlagVisibility
Visibility behavior of a flag for various user types.
- ModeratorsOnly—The flag is visible only to users with moderation permissions on the flagged element or item.
- SelfAndModerators—The flag is visible to the creator of the flag and to users with moderation permissions on the flagged element or item.

note
Type: String
A note of up to 4,000 characters about the flag.

Return Value
Type: ConnectApi.ModerationFlags

Usage
To add a flag to a comment, Allow members to flag content must be selected for an Experience Cloud site.

addFlagToComment(communityId, commentId, type, visibility, note)
Add a moderation flag of the specified type and visibility with a note to a comment.

API Version
38.0

Requires Chatter
Yes

Signature
public static ConnectApi.ModerationFlags addFlagToComment(String communityId, String commentId, ConnectApi.CommunityFlagType type, ConnectApi.CommunityFlagVisibility visibility, String note)
Parameters

communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

commentId
  Type: String
  ID for a comment.

type
  Type: ConnectApi.CommunityFlagType
  Type of moderation flag.
  • FlagAsInappropriate—Flag for inappropriate content.
  • FlagAsSpam—Flag for spam.
  If a type isn’t specified, it defaults to FlagAsInappropriate.

visibility
  Type: ConnectApi.CommunityFlagVisibility
  Visibility behavior of a flag for various user types.
  • ModeratorsOnly—The flag is visible only to users with moderation permissions on the flagged element or item.
  • SelfAndModerators—The flag is visible to the creator of the flag and to users with moderation permissions on the flagged element or item.

note
  Type: String
  A note of up to 4,000 characters about the flag.

Return Value

Type: ConnectApi.ModerationFlags

Usage

To add a flag to a comment, Allow members to flag content must be selected for an Experience Cloud site.

addFlagToFeedElement(communityId, feedElementId)

Add a moderation flag to a feed element.

API Version

31.0

Requires Chatter

Yes
Signature

```java
public static ConnectApi.ModerationCapability addFlagToFeedElement(String communityId, String feedElementId)
```

Parameters

`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

`feedElementId`
Type: `String`
ID of the feed element.

Return Value
Type: `ConnectApi.ModerationCapability`
If the feed element doesn’t support this capability, the return value is `ConnectApi.NotFoundException`.

Usage
To add a flag to a feed element, Allow members to flag content must be selected for an Experience Cloud site.

```java
addFlagToFeedElement(communityId, feedElementId, visibility)
```
Add a moderation flag of the specified visibility to a feed element.

API Version
31.0

Requires Chatter
Yes

Signature

```java
public static ConnectApi.ModerationCapability addFlagToFeedElement(String communityId, String feedElementId, ConnectApi.CommunityFlagVisibility visibility)
```

Parameters

`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

`feedElementId`
Type: `String`
ID of the feed element.
visibility
Type: ConnectApi.CommunityFlagVisibility
Visibility behavior of a flag for various user types. One of these values:
- ModeratorsOnly—The flag is visible only to users with moderation permissions on the flagged element or item.
- SelfAndModerators—The flag is visible to the creator of the flag and to users with moderation permissions on the flagged element or item.

Return Value
Type: ConnectApi.ModerationCapability
If the feed element doesn’t support this capability, the return value is ConnectApi.NotFoundException.

Usage
To add a flag to a feed element, Allow members to flag content must be selected for an Experience Cloud site.

addFlagToFeedElement(communityId, feedElementId, type)
Add a moderation flag of the specified type to a feed element.

API Version
38.0

Requires Chatter
Yes

Signature
public static ConnectApi.ModerationCapability addFlagToFeedElement(String communityId, String feedElementId, ConnectApi.CommunityFlagType type)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element.

type
Type: ConnectApi.CommunityFlagType
Type of moderation flag.
- FlagAsInappropriate—Flag for inappropriate content.
- FlagAsSpam—Flag for spam.
If a type isn’t specified, it defaults to FlagAsInappropriate.

Return Value
Type: ConnectApi.ModerationCapability

Usage
To add a flag to a feed element, Allow members to flag content must be selected for an Experience Cloud site.

`addFlagToFeedElement(communityId, feedElementId, note)`
Add a moderation flag with a note to a feed element.

API Version
38.0

Requires Chatter
Yes

Signature
`public static ConnectApi.ModerationCapability addFlagToFeedElement(String communityId, String feedElementId, String note)`

Parameters
`communityId`
Type: String
ID for an Experience Cloud site, internal, or null.

`feedElementId`
Type: String
ID of the feed element.

`note`
Type: String
A note of up to 4,000 characters about the flag.

Return Value
Type: ConnectApi.ModerationCapability

Usage
To add a flag to a feed element, Allow members to flag content must be selected for an Experience Cloud site.
addFlagToFeedElement(communityId, feedElementId, type, note)
Add a moderation flag of the specified type with a note to a feed element.

API Version
38.0

Requires Chatter
Yes

Signature
public static ConnectApi.ModerationCapability addFlagToFeedElement(String communityId, String feedElementId, ConnectApi.CommunityFlagType type, String note)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element.

type
Type: ConnectApi.CommunityFlagType
Type of moderation flag.
• FlagAsInappropriate—Flag for inappropriate content.
• FlagAsSpam—Flag for spam.
If a type isn’t specified, it defaults to FlagAsInappropriate.

note
Type: String
A note of up to 4,000 characters about the flag.

Return Value
Type: ConnectApi.ModerationCapability

Usage
To add a flag to a feed element, Allow members to flag content must be selected for an Experience Cloud site.

addFlagToFeedElement(communityId, feedElementId, type, visibility)
Add a moderation flag of the specified type and visibility to a feed element.
API Version
38.0

Requires Chatter
Yes

Signature
public static ConnectApi.ModerationCapability addFlagToFeedElement(String communityId,
String feedElementId, ConnectApi.CommunityFlagType type,
ConnectApi.CommunityFlagVisibility visibility)

Parameters

communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

feedElementId
  Type: String
  ID of the feed element.

type
  Type: ConnectApi.CommunityFlagType
  Type of moderation flag.
    • FlagAsInappropriate—Flag for inappropriate content.
    • FlagAsSpam—Flag for spam.
  If a type isn’t specified, it defaults to FlagAsInappropriate.

visibility
  Type: ConnectApi.CommunityFlagVisibility
  Visibility behavior of a flag for various user types. One of these values:
    • ModeratorsOnly—The flag is visible only to users with moderation permissions on the flagged element or item.
    • SelfAndModerators—The flag is visible to the creator of the flag and to users with moderation permissions on the flagged element or item.

Return Value
Type: ConnectApi.ModerationCapability

Usage
To add a flag to a feed element, Allow members to flag content must be selected for an Experience Cloud site.

addFlagToFeedElement (communityId, feedElementId, visibility, note)
Add a moderation flag of the specified visibility with a note to a feed element.
API Version
38.0

Requires Chatter
Yes

Signature
public static ConnectApi.ModerationCapability addFlagToFeedElement(String communityId, String feedElementId, ConnectApi.CommunityFlagVisibility visibility, String note)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element.

visibility
Type: ConnectApi.CommunityFlagVisibility
Visibility behavior of a flag for various user types. One of these values:
- ModeratorsOnly—The flag is visible only to users with moderation permissions on the flagged element or item.
- SelfAndModerators—The flag is visible to the creator of the flag and to users with moderation permissions on the flagged element or item.

note
Type: String
A note of up to 4,000 characters about the flag.

Return Value
Type: ConnectApi.ModerationCapability

Usage
To add a flag to a feed element, Allow members to flag content must be selected for an Experience Cloud site.

addFlagToFeedElement(communityId, feedElementId, type, visibility, note)
Add a moderation flag of the specified type and visibility with a note to a feed element.

API Version
38.0
Requires Chatter
Yes

Signature

```
public static ConnectApi.ModerationCapability addFlagToFeedElement(String communityId, String feedElementId, ConnectApi.CommunityFlagType type, ConnectApi.CommunityFlagVisibility visibility, String note)
```

Parameters

customId
Type: String
ID for an Experience Cloud site, internal, or null.

`feedElementId`
Type: String
ID of the feed element.

type
Type: ConnectApi.CommunityFlagType
Type of moderation flag.
- FlagAsInappropriate—Flag for inappropriate content.
- FlagAsSpam—Flag for spam.
If a type isn’t specified, it defaults to FlagAsInappropriate.

visibility
Type: ConnectApi.CommunityFlagVisibility
Visibility behavior of a flag for various user types. One of these values:
- ModeratorsOnly—The flag is visible only to users with moderation permissions on the flagged element or item.
- SelfAndModerators—The flag is visible to the creator of the flag and to users with moderation permissions on the flagged element or item.

`note`
Type: String
A note of up to 4,000 characters about the flag.

Return Value
Type: ConnectApi.ModerationCapability

Usage
To add a flag to a feed element, Allow members to flag content must be selected for an Experience Cloud site.

```
getFlagsOnComment(communityId, commentId)
```

Get the moderation flags on a comment.
API Version
29.0

Requires Chatter
Yes

Signature
public static ConnectApi.ModerationFlags getFlagsOnComment(String communityId, String commentId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

commentId
Type: String
ID for a comment.

Return Value
Type: ConnectApi.ModerationFlags

Usage
To get moderation flags, the context user must have the Moderate Experiences Feeds permission.

getFlagsOnComment(communityId, commentId, visibility)
Get the moderation flags with specified visibility on a comment.

API Version
30.0

Requires Chatter
Yes

Signature
public static ConnectApi.ModerationFlags getFlagsOnComment(String communityId, String commentId, ConnectApi.CommunityFlagVisibility visibility)
Parameters

communityId  
Type: String  
ID for an Experience Cloud site, internal, or null.

commentId  
Type: String  
ID for a comment.

visibility  
Type: ConnectApi.CommunityFlagVisibility  
Visibility behavior of a flag for various user types.
- ModeratorsOnly—The flag is visible only to users with moderation permissions on the flagged element or item.
- SelfAndModerators—The flag is visible to the creator of the flag and to users with moderation permissions on the flagged element or item.

Return Value

Type: ConnectApi.ModerationFlags

Usage

To get moderation flags, the context user must have the Moderate Experiences Feeds permission.

getFlagsOnComment(communityId, commentId, pageSize, pageParam)

Get a page of moderation flags on a comment.

API Version

40.0

Requires Chatter

Yes

Signature

public static ConnectApi.ModerationFlags getFlagsOnComment(String communityId, String commentId, Integer pageSize, String pageParam)

Parameters

communityId  
Type: String  
ID for an Experience Cloud site, internal, or null.

commentId  
Type: String
ID for a comment.

**pageSize**
Type: `Integer`  
Specifies the number of items per page. Valid values are from 1 through 100. The default size is 0.

**pageParam**  
Type: `String`  
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**Return Value**
Type: `ConnectApi.ModerationFlags`

**Usage**
To get moderation flags, the context user must have the Moderate Experiences Feeds permission.

**getFlagsOnComment(communityId, commentId, visibility, pageSize, pageParam)**
Get a page of moderation flags with specified visibility on a comment.

**API Version**
40.0

**Requires Chatter**
Yes

**Signature**
```java
public static ConnectApi.ModerationFlags getFlagsOnComment(String communityId, String commentId, ConnectApi.CommunityFlagVisibility visibility, Integer pageSize, String pageParam)
```

**Parameters**
**communityId**  
Type: `String`  
ID for an Experience Cloud site, internal, or `null`.

**commentId**  
Type: `String`  
ID for a comment.

**visibility**  
Type: `ConnectApi.CommunityFlagVisibility`  
Visibility behavior of a flag for various user types.
- **ModeratorsOnly**—The flag is visible only to users with moderation permissions on the flagged element or item.
- **SelfAndModerators**—The flag is visible to the creator of the flag and to users with moderation permissions on the flagged element or item.

**pageSize**
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. The default size is 0.

**pageParam**
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**Return Value**
Type: `ConnectApi.ModerationFlags`

**Usage**
To get moderation flags, the context user must have the Moderate Experiences Feeds permission.

**getFlagsOnFeedElement(communityId, feedElementId)**
Get the moderation flags on a feed element.

**API Version**
31.0

**Requires Chatter**
Yes

**Signature**

```java
public static ConnectApi.ModerationCapability getFlagsOnFeedElement(String communityId, String feedElementId)
```

**Parameters**

- **communityId**
  Type: String
  ID for an Experience Cloud site, `internal`, or `null`.

- **feedElementId**
  Type: String
  ID of the feed element.

**Return Value**
Type: `ConnectApi.ModerationCapability`
If the feed element doesn't support this capability, the return value is `ConnectApi.NotFoundException`. 
Usage
To get moderation flags, the context user must have the Moderate Experiences Feeds permission.

`getFlagsOnFeedElement(communityId, feedElementId, visibility)`
Get the moderation flags with specified visibility on a feed element.

API Version
31.0

Requires Chatter
Yes

Signature
```java
public static ConnectApi.ModerationCapability getFlagsOnFeedElement(String communityId, String feedElementId, ConnectApi.CommunityFlagVisibility visibility)
```

Parameters
- `communityId`: Type: String
  ID for an Experience Cloud site, internal, or null.
- `feedElementId`: Type: String
  ID of the feed element.
- `visibility`: Type: ConnectApi.CommunityFlagVisibility
  Visibility behavior of a flag for various user types. One of these values:
  - ModeratorsOnly—The flag is visible only to users with moderation permissions on the flagged element or item.
  - SelfAndModerators—The flag is visible to the creator of the flag and to users with moderation permissions on the flagged element or item.

Return Value
Type: `ConnectApi.ModerationCapability`
If the feed element doesn’t support this capability, the return value is `ConnectApi.NotFoundException`.

Usage
To get moderation flags, the context user must have the Moderate Experiences Feeds permission.

`getFlagsOnFeedElement(communityId, feedElementId, pageParam, pageSize)`
Get a page of moderation flags on a feed element.
API Version
40.0

Requires Chatter
Yes

Signature
public static ConnectApi.ModerationCapability getFlagsOnFeedElement(String communityId, String feedElementId, String pageParam, Integer pageSize)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element.

pageParam
Type: String
Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. The default size is 0.

Return Value
Type: ConnectApi.ModerationCapability
If the feed element doesn’t support this capability, the return value is ConnectApi.NotFoundException.

Usage
To get moderation flags, the context user must have the Moderate Experiences Feeds permission.

getFlagsOnFeedElement(communityId, feedElementId, visibility, pageSize, pageParam)
Get a page of moderation flags with specified visibility on a feed element.
Requires Chatter
Yes

Signature

```java
public static ConnectApi.ModerationCapability getFlagsOnFeedElement(String communityId,
String feedElementId, ConnectApi.CommunityFlagVisibility visibility, Integer pageSize,
String pageParam)
```

Parameters

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- **feedElementId**
  - Type: String
  - ID of the feed element.

- **visibility**
  - Type: `ConnectApi.CommunityFlagVisibility`
  - Visibility behavior of a flag for various user types.
    - ModeratorsOnly—The flag is visible only to users with moderation permissions on the flagged element or item.
    - SelfAndModerators—The flag is visible to the creator of the flag and to users with moderation permissions on the flagged element or item.

- **pageSize**
  - Type: Integer
  - Specifies the number of items per page. Valid values are from 1 through 100. The default size is 0.

- **pageParam**
  - Type: String
  - Page token to use to view the page. Page tokens are returned as part of the response class, for example, currentPageToken or nextPageToken. If you pass in null, the first page is returned.

Return Value

- Type: `ConnectApi.ModerationCapability`
- If the feed element doesn’t support this capability, the return value is `ConnectApi.NotFoundException`.

Usage

To get moderation flags, the context user must have the Moderate Experiences Feeds permission.

```java
removeFlagFromComment(communityId, commentId, userId)
```

Remove a moderation flag from a comment.
API Version
29.0

Requires Chatter
Yes

Signature
public static ConnectApi.ModerationFlags removeFlagFromComment(String communityId, String commentId, String userId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

commentId
Type: String
ID for a comment.

userId
Type: String
ID of the context user for whom the flag is removed. Specify null to remove the flag for all users.

Return Value
Type: Void

Usage
To remove a moderation flag, the context user must have added the flag or must have the Moderate Experiences Feeds permission.

removeFlagFromFeedElement(communityId, feedElementId, userId)
Remove a moderation flag from a feed element.

API Version
31.0

Requires Chatter
Yes

Signature
public static void removeFlagFromFeedElement(String communityId, String feedElementId, String userId)
Parameters

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- **feedElementId**
  - Type: String
  - ID of the feed element.

- **userId**
  - Type: String
  - ID of the context user for whom the flag is removed. Specify null to remove the flag for all users.

Return Value

- Type: `ConnectApi.ModerationCapability`
  - If the feed element doesn’t support this capability, the return value is `ConnectApi.NotFoundException`.

Usage

- To remove a moderation flag, the context user must have added the flag or must have the Moderate Experiences Feeds permission.

Retired CommunityModeration Methods

- The following methods for `CommunityModeration` are retired.

IN THIS SECTION:

- `addFlagToFeedItem(communityId, feedItemIds)`
  - Add a moderation flag to a feed item.

- `addFlagToFeedItem(communityId, feedItemIds, visibility)`
  - Add a moderation flag with specified visibility to a feed item.

- `getFlagsOnFeedItem(communityId, feedItemIds)`
  - Get the moderation flags on a feed item.

- `getFlagsOnFeedItem(communityId, feedItemIds, visibility)`
  - Get the moderation flags with specified visibility on a feed item.

- `removeFlagsOnFeedItem(communityId, feedItemIds, userId)`
  - Remove a moderation flag from a feed item.

- `addFlagToFeedItem(communityId, feedItemIds)`
  - Add a moderation flag to a feed item.

API Version

- 29.0–31.0
**Important:** In version 32.0 and later, use `addFlagToFeedElement(communityId, feedElementId)`.  

Requires Chatter

Yes

**Signature**

```java
public static ConnectApi.ModerationFlags addFlagToFeedItem(String communityId, String feedItemId)
```

**Parameters**

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- `feedItemId`
  - Type: `String`
  - ID for a feed item.

**Return Value**

Type: `ConnectApi.ModerationFlags`

**Usage**

To add a flag to a feed item, Allow members to flag content must be selected for an Experience Cloud site.

```java
addFlagToFeedItem(communityId, feedItemId, visibility)
```

Add a moderation flag with specified visibility to a feed item.

**API Version**

30.0–31.0

**Important:** In version 32.0 and later, use `addFlagToFeedElement(communityId, feedElementId, visibility)`.  

Requires Chatter

Yes

**Signature**

```java
public static ConnectApi.ModerationFlags addFlagToFeedItem(String communityId, String feedItemId, ConnectApi.CommunityFlagVisibility visibility)
```
Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- `feedItemId`
  - Type: `String`
  - ID for a feed item.

- `visibility`
  - Type: `ConnectApi.CommunityFlagVisibility`
  - Visibility behavior of a flag for various user types.
    - ModeratorsOnly—The flag is visible only to users with moderation permissions on the flagged element or item.
    - SelfAndModerators—The flag is visible to the creator of the flag and to users with moderation permissions on the flagged element or item.

Return Value

- Type: `ConnectApi.ModerationFlags`

Usage

To add a flag to a feed item, Allow members to flag content must be selected for an Experience Cloud site.

```java
getFlagsOnFeedItem(communityId, feedItemId)
```

Get the moderation flags on a feed item.

API Version

29.0–31.0

**Important:** In version 32.0 and later, use `getFlagsOnFeedElement(communityId, feedElementId)`.

Requires Chatter

Yes

Signature

```java
public static ConnectApi.ModerationFlags getFlagsOnFeedItem(String communityId, String feedItemId)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.
feedItemId
Type: String
ID for a feed item.

Return Value
Type: ConnectApi.ModerationFlags

Usage
To get moderation flags, the context user must have the Moderate Experiences Feeds permission.

getFlagsOnFeedItem(communityId, feedItemId, visibility)
Get the moderation flags with specified visibility on a feed item.

API Version
30.0–31.0

Important: In version 32.0 and later, use getFlagsOnFeedElement(communityId, feedElementId, visibility).

Requires Chatter
Yes

Signature
public static ConnectApi.ModerationFlags getFlagsOnFeedItem(String communityId, String feedItemId, ConnectApi.CommunityFlagVisibility visibility)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedItemId
Type: String
ID for a feed item.

visibility
Type: ConnectApi.CommunityFlagVisibility
Visibility behavior of a flag for various user types. Values are:
• ModeratorsOnly—The flag is visible only to users with moderation permissions on the flagged element or item.
• SelfAndModerators—The flag is visible to the creator of the flag and to users with moderation permissions on the flagged element or item.
Return Value
Type: ConnectApi.ModerationFlags

Usage
To get moderation flags, the context user must have the Moderate Experiences Feeds permission.

removeFlagsOnFeedItem(communityId, feedItemId, userId)
Remove a moderation flag from a feed item.

API Version
29.0–31.0

Important: In version 32.0 and later, use removeFlagFromFeedElement(communityId, feedElementId, userId).

Requires Chatter
Yes

Signature
public static ConnectApi.ModerationFlags removeFlagsOnFeedItem(String communityId, String feedItemId, String userId)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedItemId
Type: String
ID for a feed item.

userId
Type: String
ID of the context user for whom the flag is removed. Specify null to remove the flag for all users.

Return Value
Type: Void

Usage
To remove a moderation flag, the context user must have added the flag or must have the Moderate Experiences Feeds permission.
ContentHub Class

Access Files Connect repositories and their files and folders.

Namespace

ConnectApi

ContentHub Methods

The following are methods for ContentHub. All methods are static.

Use ContentHub methods to work with Files Connect repositories.

IN THIS SECTION:

- `addRepositoryItem(repositoryId, repositoryFolderId, file)`
  Add a repository item.

- `addRepositoryItem(communityId, repositoryId, repositoryFolderId, file)`
  Add a repository item in an Experience Cloud site.

- `addRepositoryItem(repositoryId, repositoryFolderId, file, fileData)`
  Add a repository item, including the binary file.

- `addRepositoryItem(communityId, repositoryId, repositoryFolderId, file, fileData)`
  Add a repository item, including the binary file, in an Experience Cloud site.

- `getAllowedItemTypes(repositoryId, repositoryFolderId)`
  Get the item types that the context user is allowed to create in the repository folder.

- `getAllowedItemTypes(repositoryId, repositoryFolderId, filter)`
  Get the item types, filtered by type, that the context user is allowed to create in the repository folder.

- `getAllowedItemTypes(communityId, repositoryId, repositoryFolderId)`
  Get the item types that the context user is allowed to create in the repository folder in an Experience Cloud site.

- `getAllowedItemTypes(communityId, repositoryId, repositoryFolderId, filter)`
  Get the item types, filtered by type, that the context user is allowed to create in the repository folder in an Experience Cloud site.

- `getFilePreview(repositoryId, repositoryFileId, formatType)`
  Get a repository file preview.

- `getFilePreview(repositoryId, repositoryFileId, formatType, startPageNumber, endPageNumber)`
  Get a page or page range of a repository file preview.

- `getFilePreview(communityId, repositoryId, repositoryFileId, formatType)`
  Get a repository file preview in an Experience Cloud site.

- `getFilePreview(communityId, repositoryId, repositoryFileId, formatType, startPageNumber, endPageNumber)`
  Get a page or page range of a repository file preview in an Experience Cloud site.

- `getItemType(repositoryId, repositoryItemTypeId)`
  Get information about an item type associated with a repository.

- `getItemType(communityId, repositoryId, repositoryItemTypeId)`
  Get information about an item type associated with a repository in an Experience Cloud site.
getPreviews(repositoryId, repositoryFileId)
Get information about a repository file’s supported previews.

getPreviews(communityId, repositoryId, repositoryFileId)
Get information about a repository file’s supported previews in an Experience Cloud site.

getRepositories()
Get a list of repositories.

getRepositories(communityId)
Get a list of repositories in an Experience Cloud site.

getRepositories(pageParam, pageSize)
Get a page of repositories.

getRepositories(communityId, pageParam, pageSize)
Get a page of repositories in an Experience Cloud site.

getRepository(repositoryId)
Get a repository.

getRepository(communityId, repositoryId)
Get a repository in an Experience Cloud site.

getRepositoryFile(repositoryId, repositoryFileId)
Get a repository file.

getRepositoryFile(repositoryId, repositoryFileId, includeExternalFilePermissionsInfo)
Get a repository file with or without permissions information.

getRepositoryFile(communityId, repositoryId, repositoryFileId)
Get a repository file in an Experience Cloud site.

getRepositoryFile(communityId, repositoryId, repositoryFileId, includeExternalFilePermissionsInfo)
Get a repository file with or without permissions information in an Experience Cloud site.

getRepositoryFolder(repositoryId, repositoryFolderId)
Get a repository folder.

getRepositoryFolder(communityId, repositoryId, repositoryFolderId)
Get a repository folder in an Experience Cloud site.

getRepositoryFolderItems(repositoryId, repositoryFolderId)
Get repository folder items.

getRepositoryFolderItems(communityId, repositoryId, repositoryFolderId)
Get repository folder items in an Experience Cloud site.

getRepositoryFolderItems(communityId, repositoryId, repositoryFolderId, pageParam, pageSize)
Get a page of repository folder items.

getRepositoryFolderItems(communityId, repositoryId, repositoryFolderId, pageParam, pageSize)
Get a page of repository folder items in an Experience Cloud site.

updateRepositoryFile(repositoryId, repositoryFileId, file)
Update the name of a repository file.

updateRepositoryFile(repositoryId, repositoryFileId, file, fileData)
Update the content of a repository file.
updateRepositoryFile(communityId, repositoryId, repositoryFileId, file)
Update the name of a repository file in an Experience Cloud site.

updateRepositoryFile(communityId, repositoryId, repositoryFileId, file, fileData)
Update the content of a repository file in an Experience Cloud site.

addRepositoryItem(repositoryId, repositoryFolderId, file)
Add a repository item.

API Version
39.0

Requires Chatter
No

Signature
public static ConnectApi.RepositoryFolderItem addRepositoryItem(String repositoryId,
String repositoryFolderId, ConnectApi.ContentHubItemInput file)

Parameters
repositoryId
Type: String
The ID of the repository.
repositoryFolderId
Type: String
The ID of the repository folder.
file
Type: ConnectApi.ContentHubItemInput
The item type ID and fields of the item type.

Return Value
Type: ConnectApi.RepositoryFolderItem

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.
Example
This example creates a file without binary content (metadata only) in a Google Drive repository folder. After the file is created, we show the file’s ID, name, description, external URL, and download URL.

```java
final String gDriveRepositoryId = '0XCxx0000000000Dgay', gDriveFolderId = 'folder:0B0lTys1KmM3sSVJ2b1zTGFqSw';

final ConnectApi.ContentHubItemInput newItem = new ConnectApi.ContentHubItemInput();
newItem.itemTypeId = 'document'; //see getAllowedTypes for any file item types available for creation/update
newItem.fields = new List<ConnectApi.ContentHubFieldValueInput>();

//Metadata: name field
final ConnectApi.ContentHubFieldValueInput fieldValueInput = new
ConnectApi.ContentHubFieldValueInput();
fieldValueInput.name = 'name';
fieldValueInput.value = 'new folder item name.txt';
newItem.fields.add(fieldValueInput);

//Metadata: description field
final ConnectApi.ContentHubFieldValueInput fieldValueInputDesc = new
ConnectApi.ContentHubFieldValueInput();
fieldValueInputDesc.name = 'description';
fieldValueInputDesc.value = 'It does describe it';
newItem.fields.add(fieldValueInputDesc);

final ConnectApi.RepositoryFolderItem newFolderItem =
ConnectApi.ContentHub.addRepositoryItem(gDriveRepositoryId, gDriveFolderId, newItem);
final ConnectApi.RepositoryFileSummary newFile = newFolderItem.file;
System.debug(String.format('New file - id: \''{0}'\', name: \''{1}'\', description: \''{2}'\', external URL: \''{3}'\', download URL: \''{4}'\'', new String[]{newFile.id, newFile.name, newFile.description, newFile.externalDocumentUrl, newFile.downloadUrl});
```

SEE ALSO:
- `setTestAddRepositoryItem(repositoryId, repositoryFolderId, file, result)`

*Apex Developer Guide: Testing ConnectApi Code*

`addRepositoryItem(communityId, repositoryId, repositoryFolderId, file)`
Add a repository item in an Experience Cloud site.

API Version
39.0

Requires Chatter
No
Signature

```java
public static ConnectApi.RepositoryFolderItem addRepositoryItem(String communityId,
String repositoryId, String repositoryFolderId, ConnectApi.ContentHubItemInput file)
```

Parameters

- `communityId`  
  Type: `String`  
  ID for an Experience Cloud site, internal, or `null`.

- `repositoryId`  
  Type: `String`  
  The ID of the repository.

- `repositoryFolderId`  
  Type: `String`  
  The ID of the repository folder.

- `file`  
  Type: `ConnectApi.ContentHubItemInput`  
  The item type ID and fields of the item type.

Return Value

Type: `ConnectApi.RepositoryFolderItem`

Usage

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- `setTestAddRepositoryItem(communityId, repositoryId, repositoryFolderId, file, result)`

  *Apex Developer Guide: Testing ConnectApi Code*

```java
addRepositoryItem(repositoryId, repositoryFolderId, file, fileData)
```

Add a repository item, including the binary file.

API Version

39.0

Requires Chatter

No
**Signature**

```java
public static ConnectApi.RepositoryFolderItem addRepositoryItem(String repositoryId,
String repositoryFolderId, ConnectApi.ContentHubItemInput file, ConnectApi.BinaryInput
fileData)
```

**Parameters**

- `repositoryId`
  - Type: `String`
  - The ID of the repository.

- `repositoryFolderId`
  - Type: `String`
  - The ID of the repository folder.

- `file`
  - Type: `ConnectApi.ContentHubItemInput`
  - The item type ID and fields of the item type.

- `fileData`
  - Type: `ConnectApi.BinaryInput`
  - The binary file.

**Return Value**

Type: `ConnectApi.RepositoryFolderItem`

**Usage**

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

**Example**

This example creates a file with binary content and metadata in a Google Drive repository folder. After the file is created, we show the file's ID, name, description, external URL, and download URL.

```java
final String gDriveRepositoryId = '0XCxx00000000ODGAY',  
gDriveFolderId = 'folder:0B01Tys1KmM3sSVJ2bjIzTGFqSWs';

final ConnectApi.ContentHubItemInput newItem = new ConnectApi.ContentHubItemInput();
newItem.itemTypeId = 'document';  //see getAllowedTypes for any file item types available
for creation/update
newItem.fields = new List<ConnectApi.ContentHubFieldValueInput>();

final String newFileName = 'new folder item name.txt';
final ConnectApi.ContentHubFieldValueInput fieldValueInput = new
ConnectApi.ContentHubFieldValueInput();
fieldValueInput.name = 'name';
fieldValueInput.value = newFileName;
newItem.fields.add(fieldValueInput);
```
//Metadata: description field
final ConnectApi.ContentHubFieldValueInput fieldValueInputDesc = new ConnectApi.ContentHubFieldValueInput();
fieldValueInputDesc.name = 'description';
fieldValueInputDesc.value = 'It does describe it';
newItem.fields.add(fieldValueInputDesc);

//Binary content
final Blob newFileBlob = Blob.valueOf('awesome content for brand new file');
final String newFileMimeType = 'text/plain';
final ConnectApi.BinaryInput fileBinaryInput = new ConnectApi.BinaryInput(newFileBlob, newFileMimeType, newFileName);

final ConnectApi.RepositoryFolderItem newFolderItem = ConnectApi.ContentHub.addRepositoryItem(gDriveRepositoryId, gDriveFolderId, newItem, fileBinaryInput);
final ConnectApi.RepositoryFileSummary newFile = newFolderItem.file;
System.debug(String.format('New file - id: "{0}", name: "{1}", description: "{2}" 
 external URL: "{3}", download URL: "{4}", new String[]{{newFile.id, newFile.name, newFile.description, newFile.externalDocumentUrl, newFile.downloadUrl}}));

SEE ALSO:
setTestAddRepositoryItem(repositoryId, repositoryFolderId, file, fileData, result)


addRepositoryItem(communityId, repositoryId, repositoryFolderId, file, fileData)
Add a repository item, including the binary file, in an Experience Cloud site.

API Version
39.0

Requires Chatter
No

Signature
public static ConnectApi.RepositoryFolderItem addRepositoryItem(String communityId, String repositoryId, String repositoryFolderId, ConnectApi.ContentHubItemInput file, ConnectApi.BinaryInput fileData)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

repositoryId
  Type: String
  The ID of the repository.

repositoryFolderId
  Type: String
  The ID of the repository folder.

file
  Type: ConnectApi.ContentHubItemInput
  The item type ID and fields of the item type.

fileData
  Type: ConnectApi.BinaryInput
  The binary file.

Return Value
  Type: ConnectApi.RepositoryFolderItem

Usage
  To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
  setTestAddRepositoryItem(communityId, repositoryId, repositoryFolderId, file, fileData, result)

getAllowedItemTypes(repositoryId, repositoryFolderId)
Get the item types that the context user is allowed to create in the repository folder.

API Version
  39.0

Requires Chatter
  No

Signature
  public static ConnectApi.ContentHubAllowedItemTypeCollection getAllowedItemTypes(String repositoryId, String repositoryFolderId)
Parameters

repositoryId
Type: String
The ID of the repository.

repositoryFolderId
Type: String
The ID of the repository folder.

Return Value
Type: ConnectApi.ContentHubAllowedItemTypeCollection

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetAllowedItemTypes(repositoryId, repositoryFolderId, result)

getAllowedItemTypes(repositoryId, repositoryFolderId, filter)
Get the item types, filtered by type, that the context user is allowed to create in the repository folder.

API Version
39.0

Requires Chatter
No

Signature
public static ConnectApi.ContentHubAllowedItemTypeCollection getAllowedItemTypes(String repositoryId, String repositoryFolderId, ConnectApi.ConnectContentHubItemType filter)

Parameters

repositoryId
Type: String
The ID of the repository.

repositoryFolderId
Type: String
The ID of the repository folder.
filter
Type: ConnectApi.ContentHubItemType
Item types. Values are:
• Any—Includes files and folders.
• FilesOnly—Includes files only.
• FoldersOnly—Includes folders only.

Return Value
Type: ConnectApi.ContentHubAllowedItemTypeCollection

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

Example
This example calls getAllowedItemTypes(repositoryId, repositoryFolderId, ConnectApi.ContentHubItemType.FilesOnly) to get the first ConnectApi.ContentHubItemTypeSummary.id of a file. The context user can create allowed files in a repository folder in the external system.

```java
final ConnectApi.ContentHubAllowedItemTypeCollection allowedItemTypesColl = ConnectApi.ContentHub.getAllowedItemTypes(repositoryId, repositoryFolderId, ConnectApi.ContentHubItemType.FilesOnly);
final List<ConnectApi.ContentHubItemTypeSummary> allowedItemTypes = allowedItemTypesColl.allowedItemTypes;
string allowedFileItemTypeIds = null;
if(allowedItemTypes.size() > 0){
    ConnectApi.ContentHubItemTypeSummary allowedItemTypeSummary = allowedItemTypes.get(0);
    allowedFileItemTypeIds = allowedItemTypeSummary.id;
}
```

SEE ALSO:
setTestGetAllowedItemTypes(repositoryId, repositoryFolderId, filter, result)

getAllowedItemTypes(communityId, repositoryId, repositoryFolderId)
Get the item types that the context user is allowed to create in the repository folder in an Experience Cloud site.

API Version
39.0

Requires Chatter
No
**Signature**

```java
public static ConnectApi.ContentHubAllowedItemTypeCollection getAllowedItemTypes(String communityId, String repositoryId, String repositoryFolderId)
```

**Parameters**

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or null.

- **repositoryId**
  - Type: `String`
  - The ID of the repository.

- **repositoryFolderId**
  - Type: `String`
  - The ID of the repository folder.

**Return Value**

Type: `ConnectApi.ContentHubAllowedItemTypeCollection`

**Usage**

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

**SEE ALSO:**

- `setTestGetAllowedItemTypes(communityId, repositoryId, repositoryFolderId, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

**getAllowedItemTypes(communityId, repositoryId, repositoryFolderId, filter)**

Get the item types, filtered by type, that the context user is allowed to create in the repository folder in an Experience Cloud site.

**API Version**

39.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.ContentHubAllowedItemTypeCollection getAllowedItemTypes(String communityId, String repositoryId, String repositoryFolderId, ConnectApi.ConnectContentHubItemType filter)
```
Parameters

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- **repositoryId**
  - Type: String
  - The ID of the repository.

- **repositoryFolderId**
  - Type: String
  - The ID of the repository folder.

- **filter**
  - Type: ConnectApi.ContentHubItemType
  - Item types. Values are:
    - **Any**—Includes files and folders.
    - **FilesOnly**—Includes files only.
    - **FoldersOnly**—Includes folders only.

Return Value

Type: ConnectApi.ContentHubAllowedItemTypeCollection

Usage

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- `setTestGetAllowedItemTypes(communityId, repositoryId, repositoryFolderId, filter, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

**getFilePreview(repositoryId, repositoryFileId, formatType)**

Get a repository file preview.

API Version

39.0

Requires Chatter

No

Signature

```java
public static ConnectApi.FilePreview getFilePreview(String repositoryId, String repositoryFileId, ConnectApi.FilePreviewFormat formatType)
```
Parameters

repositoryId
Type: String
The ID of the repository.

repositoryFileId
Type: String
The ID of the repository file.

formatType
Type: ConnectApi.FilePreviewFormat
Specifies the format of the file preview. Values are:

- Jpg—Preview format is JPG.
- Pdf—Preview format is PDF.
- Svg—Preview format is compressed SVG.
- Thumbnail—Preview format is 240 x 180 PNG.
- ThumbnailBig—Preview format is 720 x 480 PNG.
- ThumbnailTiny—Preview format is 120 x 90 PNG.

PDF previews are available for files of type DOC, DOCX, PPT, PPTX, TEXT, XLS, and XLSX. SVG files are generated on demand.

If you’re concerned that feature-rich SVG previews don’t work in your org, choose alternative file previews. To use JPG file previews, enter general in the Quick Find box in Setup. Select General Settings, and then select Display alternative file previews.

Return Value
Type: ConnectApi.FilePreview

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

Example
This example calls getFilePreview(repositoryId, repositoryFileId, ConnectApi.FilePreviewFormat.Thumbnail) to get the thumbnail format preview along with its respective URL and number of thumbnail renditions. For each thumbnail format, we show every rendition URL available.

```java
final String gDriveRepositoryId = '0XCxx00000000ODGAY', gDriveFileId = 'document:1-zcAlBaecQbo2_yNF1HCCk6QJTPmOke4KHFC4TY3rk';final ConnectApi.FilePreview filePreview = ConnectApi.ContentHub.getFilePreview(gDriveRepositoryId, gDriveFileId, ConnectApi.FilePreviewFormat.Thumbnail);System.debug(String.format('Preview - URL: \\
"\"{0}\\", format: \\
"\"{1}\\", nbr of \nrenditions for this format: {2}\n', new String[]{ filePreview.url, filePreview.format.name(),String.valueOf(filePreview.previewUrls.size())});for (ConnectApi.FilePreviewUrl filePreviewUrl : filePreview.previewUrls){
```

1156
SEE ALSO:
   setTestGetFilePreview(repositoryId, repositoryFileId, formatType, result)

getFilePreview(repositoryId, repositoryFileId, formatType, startPageNumber, endPageNumber)
Get a page or page range of a repository file preview.

API Version
39.0

Requires Chatter
No

Signature
public static ConnectApi.FilePreview getFilePreview(String repositoryId, String repositoryFileId, ConnectApi.FilePreviewFormat formatType, Integer startPageNumber, Integer endPageNumber)

Parameters
repositoryId
Type: String
The ID of the repository.

repositoryFileId
Type: String
The ID of the repository file.

formatType
Type: ConnectApi.FilePreviewFormat
Specifies the format of the file preview. Values are:
• Jpg—Preview format is JPG.
• Pdf—Preview format is PDF.
• Svg—Preview format is compressed SVG.
• Thumbnail—Preview format is 240 x 180 PNG.
• ThumbnailBig—Preview format is 720 x 480 PNG.
• ThumbnailTiny—Preview format is 120 x 90 PNG.
PDF previews are available for files of type DOC, DOCX, PPT, PPTX, TEXT, XLS, and XLSX. SVG files are generated on demand.
If you’re concerned that feature-rich SVG previews don’t work in your org, choose alternative file previews. To use JPG file previews, enter `general` in the Quick Find box in Setup. Select General Settings, and then select Display alternative file previews.

**startPageNumber**
Type: Integer
The starting page number in the range of file preview URLs.

**endPageNumber**
Type: Integer
The ending page number in the range of file preview URLs.

**Return Value**
Type: `ConnectApi.FilePreview`

**Usage**
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
`setTestGetFilePreview(repositoryId, repositoryFileId, formatType, startPageNumber, endPageNumber, result)`

*Apex Developer Guide: Testing ConnectApi Code*

**getFilePreview(communityId, repositoryId, repositoryFileId, formatType)**
Get a repository file preview in an Experience Cloud site.

**API Version**
39.0

**Requires Chatter**
No

**Signature**

```java
public static ConnectApi.FilePreview getFilePreview(String communityId, String repositoryId, String repositoryFileId, ConnectApi.FilePreviewFormat formatType)
```

**Parameters**

**communityId**
Type: String
ID for an Experience Cloud site, internal, or `null`.

**repositoryId**
Type: String
The ID of the repository.
repositoryFileId
Type: String
The ID of the repository file.

formatType
Type: ConnectApi.FilePreviewFormat
Specifies the format of the file preview. Values are:
- Jpg—Preview format is JPG.
- Pdf—Preview format is PDF.
- Svg—Preview format is compressed SVG.
- Thumbnail—Preview format is 240 x 180 PNG.
- ThumbnailBig—Preview format is 720 x 480 PNG.
- ThumbnailTiny—Preview format is 120 x 90 PNG.

PDF previews are available for files of type DOC, DOCX, PPT, PPTX, TEXT, XLS, and XLSX. SVG files are generated on demand.

If you’re concerned that feature-rich SVG previews don’t work in your org, choose alternative file previews. To use JPG file previews, enter general in the Quick Find box in Setup. Select General Settings, and then select Display alternative file previews.

Return Value
Type: ConnectApi.FilePreview

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetFilePreview(communityId, repositoryId, repositoryFileId, formatType, result)

getFilePreview(communityId, repositoryId, repositoryFileId, formatType, startPageNumber, endPageNumber)
Get a page or page range of a repository file preview in an Experience Cloud site.

API Version
39.0

Requires Chatter
No
## Signature

```java
public static ConnectApi.FilePreview getFilePreview(String communityId, String repositoryId, String repositoryFileId, ConnectApi.FilePreviewFormat formatType, Integer startPageNumber, Integer endPageNumber)
```

## Parameters

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **repositoryId**
  - Type: `String`
  - The ID of the repository.

- **repositoryFileId**
  - Type: `String`
  - The ID of the repository file.

- **formatType**
  - Type: `ConnectApi.FilePreviewFormat`
  - Specifies the format of the file preview. Values are:
    - `Jpg`—Preview format is JPG.
    - `Pdf`—Preview format is PDF.
    - `Svg`—Preview format is compressed SVG.
    - `Thumbnail`—Preview format is 240 x 180 PNG.
    - `ThumbnailBig`—Preview format is 720 x 480 PNG.
    - `ThumbnailTiny`—Preview format is 120 x 90 PNG.

  PDF previews are available for files of type DOC, DOCX, PPT, PPTX, TEXT, XLS, and XLSX. SVG files are generated on demand.

  If you’re concerned that feature-rich SVG previews don’t work in your org, choose alternative file previews. To use JPG file previews, enter `general` in the Quick Find box in Setup. Select General Settings, and then select Display alternative file previews.

- **startPageNumber**
  - Type: `Integer`
  - The starting page number in the range of file preview URLs.

- **endPageNumber**
  - Type: `Integer`
  - The ending page number in the range of file preview URLs.

## Return Value

Type: `ConnectApi.FilePreview`
Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestGetFilePreview(communityId, repositoryId, repositoryFileId, formatType, startPageNumber, endPageNumber, result)`

getItemType(repositoryId, repositoryItemTypeId)
Get information about an item type associated with a repository.

API Version
39.0

Requires Chatter
No

Signature
```
public static ConnectApi.ContentHubItemTypeDetail getItemType(String repositoryId, String repositoryItemTypeId)
```

Parameters
- `repositoryId`
  Type: `String`
  The ID of the repository.
- `repositoryItemTypeId`
  Type: `String`
  The ID of the repository item type.

Return Value
Type: `ConnectApi.ContentHubItemTypeDetail`

Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestGetItemType(repositoryId, repositoryItemTypeId, result)`
**getItemType(communityId, repositoryId, repositoryItemTypeId)**
Get information about an item type associated with a repository in an Experience Cloud site.

**API Version**
39.0

**Requires Chatter**
No

**Signature**
```java
public static ConnectApi.ContentHubItemTypeDetail getItemType(String communityId, String repositoryId, String repositoryItemTypeId)
```

**Parameters**
- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.
- **repositoryId**
  - Type: String
  - The ID of the repository.
- **repositoryItemTypeId**
  - Type: String
  - The ID of the repository item type.

**Return Value**
Type: `ConnectApi.ContentHubItemTypeDetail`

**Usage**
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestGetItemType(communityId, repositoryId, repositoryItemTypeId, result)`

**getPreviews(repositoryId, repositoryFileId)**
Get information about a repository file’s supported previews.
API Version
39.0

Requires Chatter
No

Signature
public static ConnectApi.FilePreviewCollection getPreviews(String repositoryId, String repositoryFileId)

Parameters
repositoryId
Type: String
The ID of the repository.
repositoryFileId
Type: String
The ID of the repository file.

Return Value
Type: ConnectApi.FilePreviewCollection

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

Example
This example gets all supported preview formats and their respective URLs and number of renditions. For each supported preview format, we show every rendition URL available.

```
f finally gDriveRepositoryId = '0Xcxx00000000ODGAY', gDriveFileId =
'document:1-zca1BaeoQbo2_yNFiHCcK6QJTPmOke-kHFC4TYg3r5';
final ConnectApi.ContentHub.getPreviews(gDriveRepositoryId, gDriveFileId);
for (ConnectApi.FilePreview filePreview : previewsCollection.previews)
{ System.debug(String.format('Preview - URL: \'{0}\', format: \'{1}\', nbr of
renditions for this format: {2}', new String[]{ filePreview.url,
filePreview.format.name(),String.valueOf(filePreview.previewUrls.size())}));
    for (ConnectApi.FilePreviewUrl filePreviewUrl : filePreview.previewUrls)
    { System.debug('-----> Rendition URL: ' + filePreviewUrl.previewUrl);
```
SEE ALSO:
    setTestGetPreviews(repositoryId, repositoryFileId, result)

getPreviews(communityId, repositoryId, repositoryFileId)
Get information about a repository file’s supported previews in an Experience Cloud site.

API Version
39.0

Requires Chatter
No

Signature
public static ConnectApi.FilePreviewCollection getPreviews(String communityId, String repositoryId, String repositoryFileId)

Parameters
communityId
    Type: String
    ID for an Experience Cloud site, internal, or null.
repositoryId
    Type: String
    The ID of the repository.
repositoryFileId
    Type: String
    The ID of the repository file.

Return Value
Type: ConnectApi.FilePreviewCollection
Usage

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- `setTestGetPreviews(communityId, repositoryId, repositoryFileId, result)`

**getRepositories()**

Get a list of repositories.

API Version

39.0

Requires Chatter

No

Signature

```java
public static ConnectApi.ContentHubRepositoryCollection getRepositories()
```

Return Value

Type: `ConnectApi.ContentHubRepositoryCollection`

Usage

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

Example

Example gets all repositories and gets the first SharePoint online repository found.

```java
final string sharePointOnlineProviderType = 'ContentHubSharepointOffice365';
final ConnectApi.ContentHubRepositoryCollection repositoryCollection = ConnectApi.ContentHub.getRepositories();
ConnectApi.ContentHubRepository sharePointOnlineRepository = null;
for (ConnectApi.ContentHubRepository repository : repositoryCollection.repositories){
    if(sharePointOnlineProviderType.equalsIgnoreCase(repository.providerType.type)){
        sharePointOnlineRepository = repository;
        break;
    }
```
getRepositories(communityId)
Get a list of repositories in an Experience Cloud site.

API Version
39.0

Requires Chatter
No

Signature
public static ConnectApi.ContentHubRepositoryCollection getRepositories(String communityId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

Return Value
Type: ConnectApi.ContentHubRepositoryCollection

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetRepositories(communityId, result)

getRepositories(pageParam, pageSize)
Get a page of repositories.
API Version
39.0

Requires Chatter
No

Signature
public static ConnectApi.ContentHubRepositoryCollection getRepositories(Integer pageParam, Integer pageSize)

Parameters
pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default page size is 25.

Return Value
Type: ConnectApi.ContentHubRepositoryCollection

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetRepositories(pageParam, pageSize, result)

getRepositories(communityId, pageParam, pageSize)
Get a page of repositories in an Experience Cloud site.

API Version
39.0

Requires Chatter
No
Signature

```java
public static ConnectApi.ContentHubRepositoryCollection getRepositories(String communityId, Integer pageParam, Integer pageSize)
```

Parameters

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **pageParam**
  - Type: `Integer`
  - Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

- **pageSize**
  - Type: `Integer`
  - Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default page size is 25.

Return Value

Type: `ConnectApi.ContentHubRepositoryCollection`

Usage

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- `setTestGetRepositories(communityId, pageParam, pageSize, result)`

### getRepository (repositoryId)

Get a repository.

API Version

369.0

Requires Chatter

No

Signature

```java
public static ConnectApi.ContentHubRepository getRepository(String repositoryId)
```
Parameters
repositoryId
Type: String
The ID of the repository.

Return Value
Type: ConnectApi.ContentHubRepository

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

Example
```
final string repositoryId = '0XCxx0000000123GAA';
final ConnectApi.ContentHubRepository repository = ConnectApi.ContentHub.getRepository(repositoryId);
```

SEE ALSO:
setTestGetRepository(repositoryId, result)

getRepository(communityId, repositoryId)
Get a repository in an Experience Cloud site.

API Version
39.0

Requires Chatter
No

Signature
```
public static ConnectApi.ContentHubRepository getRepository(String communityId, String repositoryId)
```

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

repositoryId
Type: String
The ID of the repository.

**Return Value**
Type: `ConnectApi.ContentHubRepository`

**Usage**
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestGetRepository(communityId, repositoryId, result)`
  *Apex Developer Guide: Testing ConnectApi Code*

**getRepositoryFile** *(repositoryId, repositoryFileId)*
Get a repository file.

**API Version**
39.0

**Requires Chatter**
No

**Signature**

```java
public static ConnectApi.RepositoryFileDetail getRepositoryFile(String repositoryId, String repositoryFileId)
```

**Parameters**

- **repositoryId**
  Type: `String`  
The ID of the repository.

- **repositoryFileId**
  Type: `String`  
The ID of the repository file.

**Return Value**
Type: `ConnectApi.RepositoryFileDetail`

**Usage**
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.
Example

```java
final String gDriveRepositoryId = '0XCxx100000000ODGAY', gDriveFileId = 'file:0B01Tys1KmM3sTmxKNjVJbWZja00';
final ConnectApi.RepositoryFileDetail file = ConnectApi.ContentHub.getRepositoryFile(gDriveRepositoryId, gDriveFileId);
System.debug(String.format('File - name: \'{0}\', size: {1}, external URL: \'{2}\',
  download URL: \'{3}\',
  new String[]{ file.name, String.valueOf(file.contentSize), file.externalDocumentUrl,
  file.downloadUrl});
```

SEE ALSO:

- `setTestGetRepositoryFile(repositoryId, repositoryFileId, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

### getRepositoryFile(repositoryId, repositoryFileId, includeExternalFilePermissionsInfo)

Get a repository file with or without permissions information.

#### API Version

39.0

#### Requires Chatter

No

#### Signature

```java
public static ConnectApi.RepositoryFileDetail getRepositoryFile(String repositoryId,
  String repositoryFileId, Boolean includeExternalFilePermissionsInfo)
```

#### Parameters

- **repositoryId**
  - Type: `String`
  - The ID of the repository.

- **repositoryFileId**
  - Type: `String`
  - The ID of the repository file.

- **includeExternalFilePermissionsInfo**
  - Type: `Boolean`
  - Specifies whether to include permission information, such as whether the file is shared and what are the available permission types.

  Managing external file permissions is supported for Google Drive, SharePoint Online, and OneDrive for Business.
Return Value

Type: `ConnectApi.RepositoryFileDetail`

Usage

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

Example

```java
final String gDriveRepositoryId = '0XCxx00000000ODGAY', gDriveFileId = 'file:0B01Tys1KmM3sTmxKN5VJbWZja00';

final ConnectApi.RepositoryFileDetail file = 
    ConnectApi.ContentHub.getRepositoryFile(gDriveRepositoryId, gDriveFileId, true);
System.debug(String.format('File - name: ''{0}'', size: {1}, external URL: ''{2}'',
    download URL: ''{3}''', new String[]{ file.name, String.valueOf(file.contentSize),
    file.externalDocumentUrl, file.downloadUrl}));
final ConnectApi.ExternalFilePermissionInformation externalFilePermInfo =
    file.externalFilePermissionInformation;

//permission types
final List<ConnectApi.ContentHubPermissionType> permissionTypes =
    externalFilePermInfo.externalFilePermissionTypes;
for (ConnectApi.ContentHubPermissionType permissionType : permissionTypes){
    System.debug(String.format('Permission type - id: ''{0}'', label: ''{1}''', new String[]{
        permissionType.id, permissionType.label}));
}

//permission groups
final List<ConnectApi.RepositoryGroupSummary> groups =
    externalFilePermInfo.repositoryPublicGroups;
for (ConnectApi.RepositoryGroupSummary ggroup : groups){
    System.debug(String.format('Group - id: ''{0}'', name: ''{1}'', type:
        ''{2}''', new String[]{
            ggroup.id, ggroup.name, ggroup.type.name()}));
}
```

SEE ALSO:

- `setTestGetRepositoryFile(repositoryId, repositoryFileId, includeExternalFilePermissionsInfo, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

`getRepositoryFile(communityId, repositoryId, repositoryFileId)`

Get a repository file in an Experience Cloud site.

API Version

39.0
Requires Chatter
No

Signature

public static ConnectApi.RepositoryFileDetail getRepositoryFile(String communityId, String repositoryId, String repositoryFileId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

repositoryId
Type: String
The ID of the repository.

repositoryFileId
Type: String
The ID of the repository file.

Return Value
Type: ConnectApi.RepositoryFileDetail

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetRepositoryFile(communityId, repositoryId, repositoryFileId, result)

getRepositoryFile(communityId, repositoryId, repositoryFileId, includeExternalFilePermissionsInfo)
Get a repository file with or without permissions information in an Experience Cloud site.

API Version
39.0

Requires Chatter
No
Signature

```java
public static ConnectApi.RepositoryFileDetail getRepositoryFile(String communityId,
String repositoryId, String repositoryFileId, Boolean includeExternalFilePermissionsInfo)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.
- `repositoryId`
  - Type: `String`
  - The ID of the repository.
- `repositoryFileId`
  - Type: `String`
  - The ID of the repository file.
- `includeExternalFilePermissionsInfo`
  - Type: `Boolean`
  - Specifies whether to include permission information, such as whether the file is shared and what are the available permission types. Managing external file permissions is supported for Google Drive, SharePoint Online, and OneDrive for Business.

Return Value

Type: `ConnectApi.RepositoryFileDetail`

Usage

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- `setTestGetRepositoryFile(communityId, repositoryId, repositoryFileId, includeExternalFilePermissionsInfo, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

```
getRepositoryFolder(repositoryId, repositoryFolderId)
```

Get a repository folder.

API Version

39.0

Requires Chatter

No
Signature

```java
public static ConnectApi.RepositoryFolderDetail getRepositoryFolder(String repositoryId, String repositoryFolderId)
```

Parameters

- `repositoryId`
  - **Type:** String
  - The ID of the repository.

- `repositoryFolderId`
  - **Type:** String
  - The ID of the repository folder.

Return Value

- **Type:** `ConnectApi.RepositoryFolderDetail`

Usage

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

Example

```java
final String gDriveRepositoryId = '0XCxx00000000ODGAY', gDriveFolderId = 'folder:0B0lTys1KmM3sSVJ2b1zTGFqSWs';
final ConnectApi.RepositoryFolderDetail folder = ConnectApi.ContentHub.getRepositoryFolder(gDriveRepositoryId, gDriveFolderId);
System.debug(String.format('Folder - name: \'{0}\', description: \'{1}\', external URL: \'{2}\', folder items URL: \'{3}\',
  new String[]{folder.name, folder.description, folder.externalFolderUrl, folder.folderItemsUrl}));
```

SEE ALSO:

- `setTestGetRepositoryFolder(repositoryId, repositoryFolderId, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

**getRepositoryFolder(communityId, repositoryId, repositoryFolderId)**

Get a repository folder in an Experience Cloud site.

**API Version**

39.0

**Requires Chatter**

No
Signature

public static ConnectApi.RepositoryFolderDetail getRepositoryFolder(String communityId, String repositoryId, String repositoryFolderId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

repositoryId
Type: String
The ID of the repository.

repositoryFolderId
Type: String
The ID of the repository folder.

Return Value
Type: ConnectApi.RepositoryFolderDetail

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetRepositoryFolder(communityId, repositoryId, repositoryFolderId, result)

getRepositoryFolderItems (repositoryId, repositoryFolderId)
Get repository folder items.

API Version
39.0

Requires Chatter
No

Signature

public static ConnectApi.RepositoryFolderItemsCollection getRepositoryFolderItems(String repositoryId, String repositoryFolderId)
Parameters

repositoryId
Type: String
The ID of the repository.

repositoryFolderId
Type: String
The ID of the repository folder.

Return Value
Type: ConnectApi.RepositoryFolderItemsCollection

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

Example
This example gets the collection of items in a repository folder. For files, we show the file's name, size, external URL, and download URL. For folders, we show the folder's name, description, and external URL.

```java
final String gDriveRepositoryId = '0XCxx00000000ODGAY', gDriveFolderId = 'folder:0B01Tys1KmM3sSVJ2bjIzTGElfSw';
final ConnectApi.ContentHub.getRepositoryFolderItems(gDriveRepositoryId, gDriveFolderId) folderItemsColl =
final List<ConnectApi.RepositoryFolderItem> folderItems = folderItemsColl.items;
System.debug('Number of items in repository folder: ' + folderItems.size());
for(ConnectApi.RepositoryFolderItem item : folderItems){
    ConnectApi.RepositoryFileSummary fileSummary = item.file;
    if(fileSummary != null){
    }else{
        ConnectApi.RepositoryFolderSummary folderSummary = item.folder;
        System.debug(String.format('Folder item - name: ' + folderSummary.name, 'description: ' + folderSummary.description));
    }
}
```

SEE ALSO:
setTestGetRepositoryFolderItems(repositoryId, repositoryFolderId, result)


getRepositoryFolderItems(communityId, repositoryId, repositoryFolderId)
Get repository folder items in an Experience Cloud site.
API Version
39.0

Requires Chatter
No

Signature
public static ConnectApi.RepositoryFolderItemsCollection getRepositoryFolderItems(String communityId, String repositoryId, String repositoryFolderId)

Parameters
communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.
repositoryId
  Type: String
  The ID of the repository.
repositoryFolderId
  Type: String
  The ID of the repository folder.

Return Value
Type: ConnectApi.RepositoryFolderItemsCollection

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetRepositoryFolderItems(communityId, repositoryId, repositoryFolderId, result)

getRepositoryFolderItems(repositoryId, repositoryFolderId, pageParam, pageSize)
Get a page of repository folder items.

API Version
39.0
Requires Chatter
No

Signature

```java
public static ConnectApi.RepositoryFolderItemsCollection getRepositoryFolderItems(String repositoryId, String repositoryFolderId, Integer pageParam, Integer pageSize)
```

Parameters

- `repositoryId`
  - Type: `String`
  - The ID of the repository.
- `repositoryFolderId`
  - Type: `String`
  - The ID of the repository folder.
- `pageParam`
  - Type: `Integer`
  - Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.
- `pageSize`
  - Type: `Integer`
  - Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default page size is 25.

Return Value

Type: `ConnectApi.RepositoryFolderItemsCollection`

Usage

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- `setTestGetRepositoryFolderItems(repositoryId, repositoryFolderId, pageParam, pageSize, result)`

getRepositoryFolderItems(communityId, repositoryId, repositoryFolderId, pageParam, pageSize)

Get a page of repository folder items in an Experience Cloud site.

API Version

39.0
Requires Chatter
No

Signature
public static ConnectApi.RepositoryFolderItemsCollection getRepositoryFolderItems(String communityId, String repositoryId, String repositoryFolderId, Integer pageParam, Integer pageSize)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.
repositoryId
Type: String
The ID of the repository.
repositoryFolderId
Type: String
The ID of the repository folder.
pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.
pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default page size is 25.

Return Value
Type: ConnectApi.RepositoryFolderItemsCollection

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetRepositoryFolderItems(communityId, repositoryId, repositoryFolderId, pageParam, pageSize, result)

updateRepositoryFile(repositoryId, repositoryFileId, file)
Update the name of a repository file.
API Version
39.0

Requires Chatter
No

Signature
```
public static ConnectApi.RepositoryFileDetail updateRepositoryFile(String repositoryId,
String repositoryFileId, ConnectApi.ContentHubItemInput file)
```

Parameters

repositoryId
Type: String
The ID of the repository.

repositoryFileId
Type: String
The ID of the repository file.

file
Type: ConnectApi.ContentHubItemInput
The item type ID and fields of the item type.
When updating the metadata of a repository file, only the name field can be updated.

Return Value
Type: ConnectApi.RepositoryFileDetail

Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

Example
This example updates the name of a file in a Google Drive repository. After the file is updated, we show the file’s ID, name, description, external URL, download URL.

```
final String gDriveRepositoryId = '0XCxx00000000ODGAY', gDriveFolderId =
  'folder:OBolTys1KmM3sSVJ2bjiItGFqSWs', gDriveFileId =
  'document:1q9OatVpcyYBR-JWzp_PhR75u1QghwFP15zhkamKrRcQ';

final ConnectApi.ContentHubItemInput updatedItem = new ConnectApi.ContentHubItemInput();
updatedItem.itemTypeId = 'document'; //see getAllowedTypes for any file item types available
for creation/update
updatedItem.fields = new List<ConnectApi.ContentHubFieldValueInput>();

//Metadata: name field
```
```
final ConnectApi.ContentHubFieldValueInput fieldValueInputName = new 
ConnectApi.ContentHubFieldValueInput();
fieldValueInputName.name = 'name';
fieldValueInputName.value = 'updated file name.txt';
updatedItem.fields.add(fieldValueInputName);

final ConnectApi.RepositoryFileDetail updatedFile = 
ConnectApi.ContentHub.updateRepositoryFile(gDriveRepositoryId, gDriveFileId, updatedItem);
System.debug(String.format('Updated file - id: \''{0}''', name: \''{1}''
, description:
\''{2}''
, external URL: \''{3}''
, download URL: \''{4}''
, new String[]{
updatedFile.id, updatedFile.name, updatedFile.description, updatedFile.externalDocumentUrl,
updatedFile.downloadUrl}}));
```

SEE ALSO:

`setTestUpdateRepositoryFile(communityId, repositoryId, repositoryFileId, file, fileData, result)`

*Apex Developer Guide: Testing ConnectApi Code*

**updateRepositoryFile(repositoryId, repositoryFileId, file, fileData)**

Update the content of a repository file.

**API Version**

39.0

**Requires Chatter**

No

**Signature**

```
public static ConnectApi.RepositoryFileDetail updateRepositoryFile(String repositoryId,
String repositoryFileId, ConnectApi.ContentHubItemInput file, ConnectApi.BinaryInput
fileData)
```

**Parameters**

**repositoryId**

Type: `String`

The ID of the repository.

**repositoryFileId**

Type: `String`

The ID of the repository file.

**file**

Type: `ConnectApi.ContentHubItemInput`

The item type ID and fields of the item type.

When updating the metadata of a repository file, only the name field can be updated.
**fileData**

Type: `ConnectApi.BinaryInput`

The binary file.

### Return Value

Type: `ConnectApi.RepositoryFileDetail`

### Usage

To test code that uses this method, use the matching `setTest` method (prefix the method name with `setTest`). Use the `setTest` method with the same parameters or the code throws an exception.

### Example

This example updates the content and name of a file in a Google Drive repository. After the file is updated, we show the file's ID, name, description, external URL, and download URL.

```java
final String gDriveRepositoryId = '0XCxx00000000ODGAY', gDriveFolderId = 'folder:0B0lTys1KmM3sSVJ2bjIzTGFSWw', gDriveFileId = 'document:1q9OatVpcyYBK-JWzp_Phr75ulQghwFP15zhkamKzRcQ';

final ConnectApi.ContentHubItemInput updatedItem = new ConnectApi.ContentHubItemInput();
updatedItem.itemTypeId = 'document'; // see getAllowedTypes for any file item types available for creation/update
updatedItem.fields = new List<ConnectApi.ContentHubFieldValueInput>();

//Metadata: name field
final ConnectApi.ContentHubFieldValueInput fieldValueInputName = new ConnectApi.ContentHubFieldValueInput();
fieldValueInputName.name = 'name';
fieldValueInputName.value = 'updated file name.txt';
updatedItem.fields.add(fieldValueInputName);

//Binary content
final Blob updatedFileBlob = Blob.valueOf('even more awesome content for updated file');
final String updatedFileMimeType = 'text/plain';
final ConnectApi.BinaryInput fileBinaryInput = new ConnectApi.BinaryInput(updatedFileBlob, updatedFileMimeType, updatedFileName);
final ConnectApi.RepositoryFileDetail updatedFile = ConnectApi.ContentHub.updateRepositoryFile(gDriveRepositoryId, gDriveFileId, updatedItem);
System.debug(String.format('Updated file - id: \''{0}''\n, name: \''{1}''\n, description: \''{2}''\n, external URL: \''{3}''\n, download URL: \''{4}''\n, new String[]{updatedFile.id, updatedFile.name, updatedFile.description, updatedFile.externalDocumentUrl, updatedFile.downloadUrl}));
```

**SEE ALSO:**

- `setTestUpdateRepositoryFile(repositoryId, repositoryFileId, file, result)`
- *Apex Developer Guide: Testing ConnectApi Code*
updateRepositoryFile(communityId, repositoryId, repositoryFileId, file)

Update the name of a repository file in an Experience Cloud site.

API Version
39.0

Requires Chatter
No

Signature
public static ConnectApi.RepositoryFileDetail updateRepositoryFile(String communityId, String repositoryId, String repositoryFileId, ConnectApi.ContentHubItemInput file)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

repositoryId
Type: String
The ID of the repository.

repositoryFileId
Type: String
The ID of the repository file.

file
Type: ConnectApi.ContentHubItemInput
The item type ID and fields of the item type.

When updating the metadata of a repository file, only the name field can be updated.

Return Value
Type: ConnectApi.RepositoryFileDetail

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestUpdateRepositoryFile(repositoryId, repositoryFileId, file, fileData, result)

updateRepositoryFile(communityId, repositoryId, repositoryFileId, file, fileData)

Update the content of a repository file in an Experience Cloud site.

API Version

39.0

Requires Chatter

No

Signature

public static ConnectApi.RepositoryFileDetail updateRepositoryFile(String communityId, String repositoryId, String repositoryFileId, ConnectApi.ContentHubItemInput file, ConnectApi.BinaryInput fileData)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

repositoryId
Type: String
The ID of the repository.

repositoryFileId
Type: String
The ID of the repository file.

file
Type: ConnectApi.ContentHubItemInput
The item type ID and fields of the item type.
When updating the metadata of a repository file, only the name field can be updated.

fileData
Type: ConnectApi.BinaryInput
The binary file.

Return Value

Type: ConnectApi.RepositoryFileDetail
Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestUpdateRepositoryFile(communityId, repositoryId, repositoryFileId, file, result)

ContentHub Test Methods
The following are the test methods for ContentHub. All methods are static.
For information about using these methods to test your ConnectApi code, see Testing ConnectApi Code.

setTestAddRepositoryItem(repositoryId, repositoryFolderId, file, result)
Register a ConnectApi.RepositoryFolderItem object to be returned when the matching
addRepositoryItem(repositoryId, repositoryFolderId, file) method is called in a test context. Use the
method with the same parameters or you receive an exception.

API Version
40.0

Signature
public static Void setTestAddRepositoryItem(String repositoryId, String
repositoryFolderId, ConnectApi.ContentHubItemInput file, ConnectApi.RepositoryFolderItem
result)

Parameters
repositoryId
Type: String
The ID of the repository.
repositoryFolderId
Type: String
The ID of the repository folder.
file
Type: ConnectApi.ContentHubItemInput
The item type ID and fields of the item type.
result
Type: ConnectApi.RepositoryFolderItem
Object containing test data.
Return Value
Type: Void

SEE ALSO:
addRepositoryItem(repositoryId, repositoryFolderId, file)
*Apex Developer Guide: Testing ConnectApi Code*

**setTestAddRepositoryItem(communityId, repositoryId, repositoryFolderId, file, result)**

Register a `ConnectApi.RepositoryFolderItem` object to be returned when the matching `addRepositoryItem(communityId, repositoryId, repositoryFolderId, file)` method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
40.0

**Signature**
```java
public static Void setTestAddRepositoryItem(String communityId, String repositoryId, String repositoryFolderId, ConnectApi.ContentHubItemInput file, ConnectApi.RepositoryFolderItem result)
```

**Parameters**
- `communityId`
  - **Type:** String
  - ID for an Experience Cloud site, internal, or null.

- `repositoryId`
  - **Type:** String
  - The ID of the repository.

- `repositoryFolderId`
  - **Type:** String
  - The ID of the repository folder.

- `file`
  - **Type:** `ConnectApi.ContentHubItemInput`
  - The item type ID and fields of the item type.

- `result`
  - **Type:** `ConnectApi.RepositoryFolderItem`
  - Object containing test data.
Return Value
Type: Void

SEE ALSO:
  - addRepositoryItem(communityId, repositoryId, repositoryFolderId, file)
  - *Apex Developer Guide: Testing ConnectApi Code*

```java
setTestAddRepositoryItem(repositoryId, repositoryFolderId, file, fileData, result)
```

Register a `ConnectApi.RepositoryFolderItem` object to be returned when the matching
`addRepositoryItem(repositoryId, repositoryFolderId, file, fileData)` method is called in a test
context. Use the method with the same parameters or you receive an exception.

API Version
40.0

Signature
```
public static Void setTestAddRepositoryItem(String repositoryId, String repositoryFolderId, ConnectApi.ContentHubItemInput file, ConnectApi.BinaryInput fileData, ConnectApi.RepositoryFolderItem result)
```

Parameters
```
repositoryId
  Type: String
  The ID of the repository.
repositoryFolderId
  Type: String
  The ID of the repository folder.

file
  Type: ConnectApi.ContentHubItemInput
  The item type ID and fields of the item type.

fileData
  Type: ConnectApi.BinaryInput
  The binary file.

result
  Type: ConnectApi.RepositoryFolderItem
  Object containing test data.
```
Return Value
Type: Void

SEE ALSO:
addRepositoryItem(repositoryId, repositoryFolderId, file, fileData)

**setTestAddRepositoryItem(communityId, repositoryId, repositoryFolderId, file, fileData, result)**

Register a `ConnectApi.RepositoryFolderItem` object to be returned when the matching `addRepositoryItem(communityId, repositoryId, repositoryFolderId, file, fileData)` method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**
40.0

**Signature**
```java
public static Void setTestAddRepositoryItem(String communityId, String repositoryId, String repositoryFolderId, ConnectApi.ContentHubItemInput file, ConnectApi.BinaryInput fileData, ConnectApi.RepositoryFolderItem result)
```

**Parameters**
- **communityId**
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.
- **repositoryId**
  Type: `String`
  The ID of the repository.
- **repositoryFolderId**
  Type: `String`
  The ID of the repository folder.
- **file**
  Type: `ConnectApi.ContentHubItemInput`
  The item type ID and fields of the item type.
- **fileData**
  Type: `ConnectApi.BinaryInput`
  The binary file.
- **result**
  Type: `ConnectApi.RepositoryFolderItem`
  Object containing test data.
Return Value
Type: Void

SEE ALSO:
- addRepositoryItem(communityId, repositoryId, repositoryFolderId, file, fileData)
  *Apex Developer Guide: Testing ConnectApi Code*

**setTestGetAllowedItemTypes(repositoryId, repositoryFolderId, result)**

Register a `ConnectApi.ContentHubAllowedItemTypeCollection` object to be returned when the matching `getAllowedItemTypes(repositoryId, repositoryFolderId)` method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**

40.0

**Signature**

```java
public static Void setTestGetAllowedItemTypes(String repositoryId, String repositoryFolderId, ConnectApi.ContentHubAllowedItemTypeCollection result)
```

**Parameters**

- `repositoryId`
  Type: `String`
  The ID of the repository.

- `repositoryFolderId`
  Type: `String`
  The ID of the repository folder.

- `result`
  Type: `ConnectApi.ContentHubAllowedItemTypeCollection`
  Object containing test data.

**Return Value**

Type: Void

SEE ALSO:
- `getAllowedItemTypes(repositoryId, repositoryFolderId)`
  *Apex Developer Guide: Testing ConnectApi Code*
setTestGetAllowedItemTypes(repositoryId, repositoryFolderId, filter, result)

Register a ConnectApi.ContentHubAllowedItemTypeCollection object to be returned when the matching getAllowedItemTypes(repositoryId, repositoryFolderId, filter) method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version

40.0

Signature

public static Void setTestGetAllowedItemTypes(String repositoryId, String repositoryFolderId, ConnectApi.ContentHubItemType filter, ConnectApi.ContentHubAllowedItemTypeCollection result)

Parameters

repositoryId
Type: String
The ID of the repository.

repositoryFolderId
Type: String
The ID of the repository folder.

filter
Type: ConnectApi.ContentHubItemType
Item types. Values are:
• Any—Includes files and folders.
• FilesOnly—Includes files only.
• FoldersOnly—Includes folders only.

result
Type: ConnectApi.ContentHubAllowedItemTypeCollection
Object containing test data.

Return Value

Type: Void

SEE ALSO:

getAllowedItemTypes(repositoryId, repositoryFolderId, filter)

**setTestGetAllowedItemTypes(communityId, repositoryId, repositoryFolderId, result)**

Register a `ConnectApi.ContentHubAllowedItemTypeCollection` object to be returned when the matching `getAllowedItemTypes(communityId, repositoryId, repositoryFolderId)` method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**

40.0

**Signature**

```java
public static Void setTestGetAllowedItemTypes(String communityId, String repositoryId, String repositoryFolderId, ConnectApi.ContentHubAllowedItemTypeCollection result)
```

**Parameters**

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.
- **repositoryId**
  - Type: `String`
  - The ID of the repository.
- **repositoryFolderId**
  - Type: `String`
  - The ID of the repository folder.
- **result**
  - Type: `ConnectApi.ContentHubAllowedItemTypeCollection`
  - Object containing test data.

**Return Value**

Type: `Void`

**SEE ALSO:**

- `getAllowedItemTypes(communityId, repositoryId, repositoryFolderId)`
- *Apex Developer Guide: Testing ConnectApi Code*

**setTestGetAllowedItemTypes(communityId, repositoryId, repositoryFolderId, filter, result)**

Register a `ConnectApi.ContentHubAllowedItemTypeCollection` object to be returned when the matching `getAllowedItemTypes(communityId, repositoryId, repositoryFolderId, filter)` method is called in a test context. Use the method with the same parameters or you receive an exception.
API Version
40.0

Signature
public static Void setTestGetAllowedItemTypes(String communityId, String repositoryId, String repositoryFolderId, ConnectApi.ContentHubItemType filter, ConnectApi.ContentHubAllowedItemTypeCollection result)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.
repositoryId
Type: String
The ID of the repository.
repositoryFolderId
Type: String
The ID of the repository folder.
filter
Type: ConnectApi.ContentHubItemType
Item types. Values are:
• Any—Includes files and folders.
• FilesOnly—Includes files only.
• FoldersOnly—Includes folders only.
result
Type: ConnectApi.ContentHubAllowedItemTypeCollection
Object containing test data.

Return Value
Type: Void

SEE ALSO:
getAllowedItemTypes(communityId, repositoryId, repositoryFolderId, filter)

setTestGetFilePreview(repositoryId, repositoryFileId, formatType, result)
Register a ConnectApi.FilePreview object to be returned when the matching getFilePreview(repositoryId, repositoryFileId, formatType) method is called in a test context. Use the method with the same parameters or you receive an exception.
API Version
40.0

Signature
public static Void setTestGetFilePreview(String repositoryId, String repositoryFileId, ConnectApi.FilePreviewFormat formatType, ConnectApi.FilePreview result)

Parameters
repositoryId
type: String
The ID of the repository.

repositoryFileId
type: String
The ID of the repository file.

formatType
type: ConnectApi.FilePreviewFormat
Specifies the format of the file preview. Values are:
- Jpg—Preview format is JPG.
- Pdf—Preview format is PDF.
- Svg—Preview format is compressed SVG.
- Thumbnail—Preview format is 240 x 180 PNG.
- ThumbnailBig—Preview format is 720 x 480 PNG.
- ThumbnailTiny—Preview format is 120 x 90 PNG.
PDF previews are available for files of type DOC, DOCX, PPT, PPTX, TEXT, XLS, and XLSX. SVG files are generated on demand.

If you’re concerned that feature-rich SVG previews don’t work in your org, choose alternative file previews. To use JPG file previews, enter general in the Quick Find box in Setup. Select General Settings, and then select Display alternative file previews.

result
type: ConnectApi.FilePreview
Object containing test data.

Return Value
Type: Void

SEE ALSO:
- getFilePreview(repositoryId, repositoryFileId, formatType)
setTestGetFilePreview(repositoryId, repositoryFileId, formatType, startPageNumber, endPageNumber, result)

Register a ConnectApi.FilePreview object to be returned when the matching getFilePreview(repositoryId, repositoryFileId, formatType, startPageNumber, endPageNumber) method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version

40.0

Signature

public static Void setTestGetFilePreview(String repositoryId, String repositoryFileId, ConnectApi.FilePreviewFormat formatType, Integer startPageNumber, Integer endPageNumber, ConnectApi.FilePreview result)

Parameters

repositoryId
Type: String
The ID of the repository.

repositoryFileId
Type: String
The ID of the repository file.

formatType
Type: ConnectApi.FilePreviewFormat
Specifies the format of the file preview. Values are:
- Jpg—Preview format is JPG.
- Pdf—Preview format is PDF.
- Svg—Preview format is compressed SVG.
- Thumbnail—Preview format is 240 x 180 PNG.
- ThumbnailBig—Preview format is 720 x 480 PNG.
- ThumbnailTiny—Preview format is 120 x 90 PNG.
PDF previews are available for files of type DOC, DOCX, PPT, PPTX, TEXT, XLS, and XLSX. SVG files are generated on demand.
If you’re concerned that feature-rich SVG previews don’t work in your org, choose alternative file previews. To use JPG file previews, enter general in the Quick Find box in Setup. Select General Settings, and then select Display alternative file previews.

startPageNumber
Type: Integer
The starting page number in the range of file preview URLs.

endPageNumber
Type: Integer
The ending page number in the range of file preview URLs.
result
   Type: `ConnectApi.FilePreview`
   Object containing test data.

Return Value
Type: Void

SEE ALSO:
   `getFilePreview(communityId, repositoryId, repositoryFileId, formatType, startPageNumber, endPageNumber)`
   *Apex Developer Guide: Testing ConnectApi Code*

`setTestGetFilePreview(communityId, repositoryId, repositoryFileId, formatType, result)`

Register a `ConnectApi.FilePreview` object to be returned when the matching `getFilePreview(communityId, repositoryId, repositoryFileId, formatType)` method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
40.0

Signature

```java
public static Void setTestGetFilePreview(String communityId, String repositoryId, String repositoryFileId, ConnectApi.FilePreviewFormat formatType, ConnectApi.FilePreview result)
```

Parameters

- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or null.

- `repositoryId`
  Type: `String`
  The ID of the repository.

- `repositoryFileId`
  Type: `String`
  The ID of the repository file.

- `formatType`
  Type: `ConnectApi.FilePreviewFormat`
  Specifies the format of the file preview. Values are:
  - `Jpg`—Preview format is JPG.
  - `Pdf`—Preview format is PDF.
• **Svg**—Preview format is compressed SVG.
• **Thumbnail**—Preview format is 240 x 180 PNG.
• **ThumbnailBig**—Preview format is 720 x 480 PNG.
• **ThumbnailTiny**—Preview format is 120 x 90 PNG.

PDF previews are available for files of type DOC, DOCX, PPT, PPTX, TEXT, XLS, and XLSX. SVG files are generated on demand.

If you’re concerned that feature-rich SVG previews don’t work in your org, choose alternative file previews. To use JPG file previews, enter `general` in the Quick Find box in Setup. Select General Settings, and then select **Display alternative file previews**.

**result**
- Type: `ConnectApi.FilePreview`
- Object containing test data.

**Return Value**
- Type: Void

**SEE ALSO:**
- `getFilePreview(communityId, repositoryId, repositoryFileId, formatType)`
- *Apex Developer Guide: Testing ConnectApi Code*

```java
setTestGetFilePreview(communityId, repositoryId, repositoryFileId, formatType, startPageNumber, endPageNumber, result)
```

Register a `ConnectApi.FilePreview` object to be returned when the matching `getFilePreview(communityId, repositoryId, repositoryFileId, formatType, startPageNumber, endPageNumber)` method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**
- 40.0

**Signature**

```java
public static Void setTestGetFilePreview(String communityId, String repositoryId, String repositoryFileId, ConnectApi.FilePreviewFormat formatType, Integer startPageNumber, Integer endPageNumber, ConnectApi.FilePreview result)
```

**Parameters**

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **repositoryId**
  - Type: `String`
  - The ID of the repository.
repositoryFileId
   Type: String
   The ID of the repository file.

formatType
   Type: ConnectApi.FilePreviewFormat
   Specifies the format of the file preview. Values are:
   • Jpg—Preview format is JPG.
   • Pdf—Preview format is PDF.
   • Svg—Preview format is compressed SVG.
   • Thumbnail—Preview format is 240 x 180 PNG.
   • ThumbnailBig—Preview format is 720 x 480 PNG.
   • ThumbnailTiny—Preview format is 120 x 90 PNG.
   PDF previews are available for files of type DOC, DOCX, PPT, PPTX, TEXT, XLS, and XLSX. SVG files are generated on demand.
   If you’re concerned that feature-rich SVG previews don’t work in your org, choose alternative file previews. To use JPG file previews, enter general in the Quick Find box in Setup. Select General Settings, and then select Display alternative file previews.

startPageNumber
   Type: Integer
   The starting page number in the range of file preview URLs.

d_endPageNumber
   Type: Integer
   The ending page number in the range of file preview URLs.

result
   Type: ConnectApi.FilePreview
   Object containing test data.

Return Value
   Type: Void

SEE ALSO:
   getFilePreview(communityId, repositoryId, repositoryFileId, formatType, startPageNumber, endPageNumber)

setTestGetType(repositoryId, repositoryItemType, result)
   Register a ConnectApi.ContentHubItemTypeDetail object to be returned when the matching
   getItemType(repositoryId, repositoryItemType) method is called in a test context. Use the method with the
   same parameters or you receive an exception.

API Version
   40.0
Signature

```java
public static Void setTestGetItemType(String repositoryId, String repositoryItemTypeId, ConnectApi.ContentHubItemTypeDetail result)
```

Parameters

- `repositoryId`
  Type: String
  The ID of the repository.

- `repositoryItemTypeId`
  Type: String
  The ID of the repository item type.

- `result`
  Type: `ConnectApi.ContentHubItemTypeDetail`
  Object containing test data.

Return Value

Type: Void

SEE ALSO:

- `getItemType(repositoryId, repositoryItemTypeId)`
- **Apex Developer Guide: Testing ConnectApi Code**

### setTestGetItemType(communityId, repositoryId, repositoryItemTypeId, result)

Register a `ConnectApi.ContentHubItemTypeDetail` object to be returned when the matching `getItemType(communityId, repositoryId, repositoryItemTypeId)` method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version

40.0

Signature

```java
public static Void setTestGetItemType(String communityId, String repositoryId, String repositoryItemTypeId, ConnectApi.ContentHubItemTypeDetail result)
```

Parameters

- `communityId`
  Type: String
  ID for an Experience Cloud site, internal, or `null`.

- `repositoryId`
  Type: String
The ID of the repository.
repositoryItemTypeId
Type: String
The ID of the repository item type.
result
Type: ConnectApi.ContentHubItemTypeDetail
Object containing test data.

Return Value
Type: Void

SEE ALSO:
getItemType(communityId, repositoryId, repositoryItemTypeId)

setTestGetPreviews(repositoryId, repositoryFileId, result)
Register a ConnectApi.FilePreviewCollection object to be returned when the matching
getPreviews(repositoryId, repositoryFileId) method is called in a test context. Use the method with the same
parameters or you receive an exception.

API Version
40.0

Signature
public static Void setTestGetPreviews(String repositoryId, String repositoryFileId,
ConnectApi.FilePreviewCollection result)

Parameters
repositoryId
Type: String
The ID of the repository.
repositoryFileId
Type: String
The ID of the repository file.
result
Type: ConnectApi.FilePreviewCollection
Object containing test data.
Return Value
Type: Void

SEE ALSO:
getPreviews(repositoryId, repositoryFileId)

**setTestGetPreviews(communityId, repositoryId, repositoryFileId, result)**

Register a `ConnectApi.FilePreviewCollection` object to be returned when the matching `getPreviews(communityId, repositoryId, repositoryFileId)` method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
40.0

Signature

```java
public static Void setTestGetPreviews(String communityId, String repositoryId, String repositoryFileId, ConnectApi.FilePreviewCollection result)
```

Parameters

- **communityId**
  Type: String
  ID for an Experience Cloud site, internal, or null.

- **repositoryId**
  Type: String
  The ID of the repository.

- **repositoryFileId**
  Type: String
  The ID of the repository file.

- **result**
  Type: `ConnectApi.FilePreviewCollection`
  Object containing test data.

Return Value
Type: Void

SEE ALSO:
getPreviews(communityId, repositoryId, repositoryFileId)
**setTestGetRepositories (result)**

Register a `ConnectApi.ContentHubRepositoryCollection` object to be returned when the matching `getRepositories()` method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**

40.0

**Signature**

```java
public static Void setTestGetRepositories(ConnectApi.ContentHubRepositoryCollection result)
```

**Parameters**

- `result`
  - Type: `ConnectApi.ContentHubRepositoryCollection`
  - Object containing test data.

**Return Value**

Type: Void

**SEE ALSO:**
- `getRepositories()`
- *Apex Developer Guide: Testing ConnectApi Code*

**setTestGetRepositories (communityId, result)**

Register a `getRepositories(communityId)` object to be returned when the matching `ConnectApi.ContentHubRepositoryCollection` method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**

40.0

**Signature**

```java
public static Void setTestGetRepositories(String communityId, ConnectApi.ContentHubRepositoryCollection result)
```

**Parameters**

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, `internal`, or `null`.
result
   Type: ConnectApi.ContentHubRepositoryCollection
   Object containing test data.

Return Value
Type: Void

SEE ALSO:
   getRepositories(communityId)

setTestGetRepositories(pageParam, pageSize, result)
Register a ConnectApi.ContentHubRepositoryCollection object to be returned when the matching getRepositories(pageParam, pageSize) method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
40.0

Signature
public static Void setTestGetRepositories(Integer pageParam, Integer pageSize, ConnectApi.ContentHubRepositoryCollection result)

Parameters
pageParam
   Type: Integer
   Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
   Type: Integer
   Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default page size is 25.

result
   Type: ConnectApi.ContentHubRepositoryCollection
   Object containing test data.

Return Value
Type: Void

SEE ALSO:
   getRepositories(pageParam, pageSize)
**setTestGetRepositories(communityId, pageParam, pageSize, result)**

Register a `ConnectApi.ContentHubRepositoryCollection` object to be returned when the matching `getRepositories(communityId, pageParam, pageSize)` method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**

40.0

**Signature**

```java
public static Void setTestGetRepositories(String communityId, Integer pageParam, Integer pageSize, ConnectApi.ContentHubRepositoryCollection result)
```

**Parameters**

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, `internal`, or `null`.

- **pageParam**
  - Type: `Integer`
  - Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

- **pageSize**
  - Type: `Integer`
  - Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default page size is 25.

- **result**
  - Type: `ConnectApi.ContentHubRepositoryCollection`
  - Object containing test data.

**Return Value**

Type: Void

**SEE ALSO:**

- `getRepositories(communityId, pageParam, pageSize)`
- *Apex Developer Guide: Testing ConnectApi Code*

**setTestGetRepository(repositoryId, result)**

Register a `ConnectApi.ContentHubRepository` object to be returned when the matching `getRepository(repositoryId)` method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**

40.0
Signature

```java
public static Void setTestGetRepository(String repositoryId, ConnectApi.ContentHubRepository result)
```

Parameters

- `repositoryId`  
  Type: `String`  
  The ID of the repository.

- `result`  
  Type: `ConnectApi.ContentHubRepository`  
  Object containing test data.

Return Value

Type: Void

SEE ALSO:

- `getRepository(repositoryId)`

**setTestGetRepository(communityId, repositoryId, result)**

Register a `ConnectApi.ContentHubRepository` object to be returned when the matching `getRepository(communityId, repositoryId)` method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version

40.0

Signature

```java
public static Void setTestGetRepository(String communityId, String repositoryId, ConnectApi.ContentHubRepository result)
```

Parameters

- `communityId`  
  Type: `String`  
  ID for an Experience Cloud site, internal, or null.

- `repositoryId`  
  Type: `String`  
  The ID of the repository.

- `result`  
  Type: `ConnectApi.ContentHubRepository`
setTestGetRepositoryFile(repositoryId, repositoryFileId, result)

Register a ConnectApi.RepositoryFileDetail object to be returned when the matching getRepositoryFile(repositoryId, repositoryFileId) method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
40.0

Signature
public static Void setTestGetRepositoryFile(String repositoryId, String repositoryFileId, ConnectApi.RepositoryFileDetail result)

Parameters
repositoryId
Type: String
The ID of the repository.
repositoryFileId
Type: String
The ID of the repository file.
result
Type: ConnectApi.RepositoryFileDetail
Object containing test data.

Return Value
Type: Void

SEE ALSO:
getRepository(communityId, repositoryId)
**setTestGetRepositoryFile***(repositoryId, repositoryFileId, includeExternalFilePermissionsInfo, result)***

Register a ConnectApi.RepositoryFileDetail object to be returned when the matching getRepositoryFile(repositoryId, repositoryFileId, includeExternalFilePermissionsInfo) method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**

40.0

**Signature**

```
public static Void setTestGetRepositoryFile(String repositoryId, String repositoryFileId, Boolean includeExternalFilePermissionsInfo, ConnectApi.RepositoryFileDetail result)
```

**Parameters**

- **repositoryId**
  
  Type: String
  
  The ID of the repository.

- **repositoryFileId**
  
  Type: String
  
  The ID of the repository file.

- **includeExternalFilePermissionsInfo**
  
  Type: Boolean
  
  Specifies whether to include permission information, such as whether the file is shared and what are the available permission types. Managing external file permissions is supported for Google Drive, SharePoint Online, and OneDrive for Business.

- **result**
  
  Type: ConnectApi.RepositoryFileDetail
  
  Object containing test data.

**Return Value**

Type: Void

SEE ALSO:

- getRepositoryFile(repositoryId, repositoryFileId, includeExternalFilePermissionsInfo)
- *Apex Developer Guide: Testing ConnectApi Code*

**setTestGetRepositoryFile***(communityId, repositoryId, repositoryFileId, result)***

Register a ConnectApi.RepositoryFileDetail object to be returned when the matching getRepositoryFile(communityId, repositoryId, repositoryFileId) method is called in a test context. Use the method with the same parameters or you receive an exception.
API Version
40.0

Signature
public static Void setTestGetRepositoryFile(String communityId, String repositoryId, String repositoryFileId, ConnectApi.RepositoryFileDetail result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

repositoryId
Type: String
The ID of the repository.

repositoryFileId
Type: String
The ID of the repository file.

result
Type: ConnectApi.RepositoryFileDetail
Object containing test data.

Return Value
Type: Void

SEE ALSO:
getRepositoryFile(communityId, repositoryId, repositoryFileId)

setTestGetRepositoryFile(communityId, repositoryId, repositoryFileId, includeExternalFilePermissionsInfo, result)

Register a ConnectApi.RepositoryFileDetail object to be returned when the matching getRepositoryFile(communityId, repositoryId, repositoryFileId, includeExternalFilePermissionsInfo) method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
40.0
Signature

public static Void setTestGetRepositoryFile(String communityId, String repositoryId, String repositoryFileId, Boolean includeExternalFilePermissionsInfo, ConnectApi.RepositoryFileDetail result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

repositoryId
Type: String
The ID of the repository.

repositoryFileId
Type: String
The ID of the repository file.

includeExternalFilePermissionsInfo
Type: Boolean
Specifies whether to include permission information, such as whether the file is shared and what are the available permission types. Managing external file permissions is supported for Google Drive, SharePoint Online, and OneDrive for Business.

result
Type: ConnectApi.RepositoryFileDetail
Object containing test data.

Return Value
Type: Void

SEE ALSO:
getRepositoryFile(communityId, repositoryId, repositoryFileId, includeExternalFilePermissionsInfo)

setTestGetRepositoryFolder(repositoryId, repositoryFolderId, result)

Register a ConnectApi.RepositoryFolderDetail object to be returned when the matching getRepositoryFolder(repositoryId, repositoryFolderId) method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
40.0
Signature

public static Void setTestGetRepositoryFolder(String repositoryId, String repositoryFolderId, ConnectApi.RepositoryFolderDetail result)

Parameters

repositoryId
Type: String
The ID of the repository.

repositoryFolderId
Type: String
The ID of the repository folder.

result
Type: ConnectApi.RepositoryFolderDetail
Object containing test data.

Return Value

Type: Void

SEE ALSO:

getRepositoryFolder(repositoryId, repositoryFolderId)


setTestGetRepositoryFolder(communityId, repositoryId, repositoryFolderId, result)

Register a ConnectApi.RepositoryFolderDetail object to be returned when the matching getRepositoryFolder (communityId, repositoryId, repositoryFolderId) method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version

40.0

Signature

public static Void setTestGetRepositoryFolder(String communityId, String repositoryId, String repositoryFolderId, ConnectApi.RepositoryFolderDetail result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.
repositoryId
   Type: String
   The ID of the repository.
repositoryFolderId
   Type: String
   The ID of the repository folder.
result
   Type: ConnectApi.RepositoryFolderDetail
   Object containing test data.

Return Value
Type: Void

SEE ALSO:
   getRepositoryFolder(communityId, repositoryId, repositoryFolderId)

setTestGetRepositoryFolderItems(repositoryId, repositoryFolderId, result)
Register a ConnectApi.RepositoryFolderItemsCollection object to be returned when the matching
getRepositoryFolderItems(repositoryId, repositoryFolderId) method is called in a test context. Use the
method with the same parameters or you receive an exception.

API Version
40.0

Signature
public static Void setTestGetRepositoryFolderItems(String repositoryId, String
repositoryFolderId, ConnectApi.RepositoryFolderItemsCollection result)

Parameters
repositoryId
   Type: String
   The ID of the repository.
repositoryFolderId
   Type: String
   The ID of the repository folder.
result
   Type: ConnectApi.RepositoryFolderItemsCollection
   Object containing test data.
Return Value
Type: Void

SEE ALSO:
getRepositoryFolderItems(repositoryId, repositoryFolderId)


setTestGetRepositoryFolderItems(communityId, repositoryId, repositoryFolderId, result)

Register a ConnectApi.RepositoryFolderItemsCollection object to be returned when the matching getRepositoryFolderItems(communityId, repositoryId, repositoryFolderId) method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
40.0

Signature
public static Void setTestGetRepositoryFolderItems(String communityId, String repositoryId, String repositoryFolderId, ConnectApi.RepositoryFolderItemsCollection result)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

repositoryId
Type: String
The ID of the repository.

repositoryFolderId
Type: String
The ID of the repository folder.

result
Type: ConnectApi.RepositoryFolderItemsCollection
Object containing test data.
Return Value
Type: Void

SEE ALSO:
getRepositoryFolderItems(communityId, repositoryId, repositoryFolderId)

_Apex Developer Guide: Testing ConnectApi Code_

```java
setTestGetRepositoryFolderItems(repositoryId, repositoryFolderId, pageParam, pageSize, result)
```

Register a `ConnectApi.RepositoryFolderItemsCollection` object to be returned when the matching `getRepositoryFolderItems(repositoryId, repositoryFolderId, pageParam, pageSize)` method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
40.0

Signature

```java
public static Void setTestGetRepositoryFolderItems(String repositoryId, String repositoryFolderId, Integer pageParam, Integer pageSize, ConnectApi.RepositoryFolderItemsCollection result)
```

Parameters

- **repositoryId**
  - Type: `String`
  - The ID of the repository.

- **repositoryFolderId**
  - Type: `String`
  - The ID of the repository folder.

- **pageParam**
  - Type: `Integer`
  - Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

- **pageSize**
  - Type: `Integer`
  - Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default page size is 25.

- **result**
  - Type: `ConnectApi.RepositoryFolderItemsCollection`
  - Object containing test data.
Return Value
Type: Void

SEE ALSO:
getRepositoryFolderItems(repositoryId, repositoryFolderId, pageParam, pageSize)
_Apex Developer Guide: Testing ConnectApi Code_

**setTestGetRepositoryFolderItems(communityId, repositoryId, repositoryFolderId, pageParam, pageSize, result)**

Register a `ConnectApi.RepositoryFolderItemsCollection` object to be returned when the matching `getRepositoryFolderItems(communityId, repositoryId, repositoryFolderId, pageParam, pageSize)` method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
40.0

Signature

```java
class Apex Developer Guide
public static Void setTestGetRepositoryFolderItems(String communityId, String repositoryId, String repositoryFolderId, Integer pageParam, Integer pageSize, ConnectApi.RepositoryFolderItemsCollection result)
```

Parameters

- `communityId`
  Type: String
  ID for an Experience Cloud site, internal, or null.

- `repositoryId`
  Type: String
  The ID of the repository.

- `repositoryFolderId`
  Type: String
  The ID of the repository folder.

- `pageParam`
  Type: Integer
  Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

- `pageSize`
  Type: Integer
  Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default page size is 25.

- `result`
  Type: `ConnectApi.RepositoryFolderItemsCollection`
  Object containing test data.
**Return Value**

Type: Void

SEE ALSO:

getRepositoryFolderItems(communityId, repositoryId, repositoryFolderId, pageParam, pageSize)

_Apex Developer Guide: Testing ConnectApi Code_

**setTestUpdateRepositoryFile(communityId, repositoryId, repositoryFileId, file, fileData, result)**

Register a `ConnectApi.RepositoryFileDetail` object to be returned when the matching `updateRepositoryFile(communityId, repositoryId, repositoryFileId, file, fileData)` method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**

40.0

**Signature**

```java
public static Void setTestUpdateRepositoryFile(String communityId, String repositoryId, String repositoryFileId, ConnectApi.ContentHubItemInput file, ConnectApi.BinaryInput fileData, ConnectApi.RepositoryFileDetail result)
```

**Parameters**

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or null.

- **repositoryId**
  - Type: `String`
  - The ID of the repository.

- **repositoryFileId**
  - Type: `String`
  - The ID of the repository file.

- **file**
  - Type: `ConnectApi.ContentHubItemInput`
  - The item type ID and fields of the item type.

- **fileData**
  - Type: `ConnectApi.BinaryInput`
  - The binary file.

- **result**
  - Type: `ConnectApi.RepositoryFileDetail`
  - Object containing test data.
Return Value
Type: Void

SEE ALSO:
updateRepositoryFile(repositoryId, repositoryFileId, file)

setTestUpdateRepositoryFile(repositoryId, repositoryFileId, file, result)
Register a ConnectApi.RepositoryFileDetail object to be returned when the matching
updateRepositoryFile(repositoryId, repositoryFileId, file) method is called in a test context. Use the
method with the same parameters or you receive an exception.

API Version
40.0

Signature
public static Void setTestUpdateRepositoryFile(String repositoryId, String
repositoryFileId, ConnectApi.ContentHubItemInput file, ConnectApi.RepositoryFileDetail
result)

Parameters
repositoryId
Type: String
The ID of the repository.

repositoryFileId
Type: String
The ID of the repository file.

file
Type: ConnectApi.ContentHubItemInput
The item type ID and fields of the item type.

result
Type: ConnectApi.RepositoryFileDetail
Object containing test data.

Return Value
Type: Void

SEE ALSO:
updateRepositoryFile(repositoryId, repositoryFileId, file, fileData)
**setTestUpdateRepositoryFile** (repositoryId, repositoryFileId, file, fileData, result)

Register a `ConnectApi.RepositoryFileDetail` object to be returned when the matching `updateRepositoryFile` method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**

40.0

**Signature**

```java
public static Void setTestUpdateRepositoryFile(String repositoryId, String repositoryFileId, ConnectApi.ContentHubItemInput file, ConnectApi.BinaryInput fileData, ConnectApi.RepositoryFileDetail result)
```

**Parameters**

- **repositoryId**
  - Type: `String`
  - The ID of the repository.

- **repositoryFileId**
  - Type: `String`
  - The ID of the repository file.

- **file**
  - Type: `ConnectApi.ContentHubItemInput`
  - The item type ID and fields of the item type.

- **fileData**
  - Type: `ConnectApi.BinaryInput`
  - The binary file.

- **result**
  - Type: `ConnectApi.RepositoryFileDetail`
  - Object containing test data.

**Return Value**

Type: Void

**SEE ALSO:**

- `updateRepositoryFile(communityId, repositoryId, repositoryFileId, file)`
- *Apex Developer Guide: Testing ConnectApi Code*
`setTestUpdateRepositoryFile(communityId, repositoryId, repositoryFileId, file, result)`

Register a `ConnectApi.RepositoryFileDetail` object to be returned when the matching
`updateRepositoryFile(communityId, repositoryId, repositoryFileId, file)` method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**

40.0

**Signature**

```java
public static Void setTestUpdateRepositoryFile(String communityId, String repositoryId, String repositoryFileId, ConnectApi.ContentHubItemInput file, ConnectApi.RepositoryFileDetail result)
```

**Parameters**

`communityId`

- Type: `String`
- ID for an Experience Cloud site, internal, or `null`.

`repositoryId`

- Type: `String`
- The ID of the repository.

`repositoryFileId`

- Type: `String`
- The ID of the repository file.

`file`

- Type: `ConnectApi.ContentHubItemInput`
- The item type ID and fields of the item type.

`result`

- Type: `ConnectApi.RepositoryFileDetail`
- Object containing test data.

**Return Value**

Type: `Void`

**SEE ALSO:**

`updateRepositoryFile(communityId, repositoryId, repositoryFileId, file, fileData)`

*Apex Developer Guide: Testing ConnectApi Code*

**ConversationApplicationDefinition Class**

Access information about a conversation application definition.
Namespace

ConnectApi

ConversationApplicationDefinition Methods

The following are methods for ConversationApplicationDefinition. All methods are static.

IN THIS SECTION:

   getConversationApplicationDefinition(integrationName)
       Get information about an integration’s conversation application definition and the associated bot.

getConversationApplicationDefinition(integrationName)

Get information about an integration’s conversation application definition and the associated bot.

API Version

54.0

Requires Chatter

No

Signature

public static ConnectApi.ConversationApplicationDefinitionDetailRespresentation getConversationApplicationDefinition(String integrationName)

Parameters

integrationName
   Type: String
   Name of the conversation application.

Return Value

Type: ConnectApi.ConversationApplicationDefinitionDetailRespresentation

Usage

To access this method, enable the bot feature, and the user must be an admin or have the Manage Bots or Manage Bots Training Data user permissions.

Datacloud Class

Purchase Data.com contact or company records, and retrieve purchase information.
Namespace

ConnectApi

Note: When your Data.com Prospector or Data.com Clean contract expires, Data.com features, objects, and fields will be removed from your org.

To support customers’ needs around compliance and to remain a leader in trust and privacy, Salesforce removed all contact data from the Data.com service on February 1, 2021.

For more information, see Data.com Prospector and Clean Retirement.

IN THIS SECTION:

Datacloud Methods
The following are methods for Datacloud. All methods are static.

Datacloud Methods
The following are methods for Datacloud. All methods are static.

IN THIS SECTION:

getCompaniesFromOrder(orderId, pageSize, page)
Get a list of purchased company records for an order.

getCompany(companyId)
Get a company record.

getchContact(contactId)
Get a contact.

getContactsFromOrder(orderId, page, pageSize)
Get a list of purchased contacts for an order.

getAddress(orderId)
Get an order.

getUsage(userId)
Get purchase usage information for a user.

postOrder(orderInput)
Purchase records that are listed in an input file.

getCompaniesFromOrder(orderId, pageSize, page)
Get a list of purchased company records for an order.

API Version
32.0

Requires Chatter
No
Signature

public static ConnectApi.DatacloudCompanies getCompaniesFromOrder(String orderId, String pageSize, String page)

Parameters

orderId
Type: String
ID of an order.

page
Type: Integer
Number of the page that you want returned.

pageSize
Type: Integer
Number of companies to show on a page. The default pageSize is 25.

Return Value
Type: ConnectApi.DatacloudCompanies

getCompany (companyId)
Get a company record.

API Version
32.0

Requires Chatter
No

Signature

public static ConnectApi.DatacloudCompany getCompany(String companyId)

Parameters

companyId
Type: String
ID of a company in the Data.com database.

Return Value
Type: ConnectApi.DatacloudCompany
**getContact(contactId)**
Get a contact.

**API Version**
32.0

**Requires Chatter**
No

**Signature**
```
public static ConnectApi.DatacloudContact getContact(String contactId)
```

**Parameters**
- **contactId**
  Type: String
  ID of a contact in the Data.com database.

**Return Value**
Type: `ConnectApi.DatacloudContact`

**getContactsFromOrder(orderId, page, pageSize)**
Get a list of purchased contacts for an order.

**API Version**
32.0

**Requires Chatter**
No

**Signature**
```
public static ConnectApi.DatacloudContacts getContactsFromOrder(String orderId, String page, String pageSize)
```

**Parameters**
- **orderId**
  Type: String
  ID of an order.
- **page**
  Type: Integer
Number of the page that you want returned.

`pageSize`
Type: `Integer`
Number of contacts to show on a page. The default `pageSize` is 25.

Return Value
Type: `ConnectApi.DatacloudContacts`

`getOrder(orderId)`
Get an order.

API Version
32.0

Requires Chatter
No

Signature
```java
public static ConnectApi.DatacloudOrder getOrder(String orderId)
```

Parameters
`orderId`
Type: `String`
ID of an order.

Return Value
Type: `ConnectApi.DatacloudOrder`

`getUsage(userId)`
Get purchase usage information for a user.

API Version
32.0

Requires Chatter
No

Signature
```java
public static ConnectApi.DatacloudPurchaseUsage getUsage(String userId)
```
Parameters

userId
  Type: String
  ID of a user.

Return Value

Type: ConnectApi.DatacloudPurchaseUsage

**postOrder(orderInput)**

Purchase records that are listed in an input file.

API Version

32.0

Requires Chatter

No

Signature

```java
public static ConnectApi.DatacloudOrder postOrder(ConnectApi.DatacloudOrderInput orderInput)
```

Parameters

**orderInput**

  Type: ConnectApi.DatacloudOrderInput

  A list that contains IDs for the contacts or companies that you want to see.

Return Value

Type: ConnectApi.DatacloudOrder

Example

```java
ConnectApi.DatacloudOrderInput inputOrder=new ConnectApi.DatacloudOrderInput();
List<String> ids=new List<String>();
ids.add('1234');
inputOrder.companyIds=ids;
ConnectApi.DatacloudOrder datacloudOrderRep = ConnectApi.Datacloud.postOrder(inputOrder);
```

**EmailMergeFieldService Class**

Extract a list of merge fields for an object. A merge field is a field you can put in an email template, mail merge template, custom link, or formula to incorporate values from a record.
Namespace

ConnectApi

EmailMergeFieldService Methods

The following are methods for EmailMergeFieldService. All methods are static.

IN THIS SECTION:

getMergeFields(objectApiNames)

Extract the merge fields for a specific object.

**getMergeFields (objectApiNames)**

Extract the merge fields for a specific object.

API Version

39.0

Requires Chatter

No

Signature

public static ConnectApi.EmailMergeFieldInfo getMergeFields(List<String> objectApiNames)

Parameters

objectApiNames

Type: List<String>

The API names for the objects being referenced.

Return Value

Type: ConnectApi.EmailMergeFieldInfo

EmployeeProfiles Class

Get, set and crop, and delete employee banner photos and photos.

Namespace

ConnectApi
EmployeeProfiles Methods

These methods are for EmployeeProfiles. All methods are static.

IN THIS SECTION:

- deleteBannerPhoto(employeeId)
  Delete an employee’s banner photo.
- deletePhoto(employeeId)
  Delete an employee’s photo.
- getBannerPhoto(employeeId)
  Get an employee’s banner photo.
- getPhoto(employeeId)
  Get an employee’s photo.
- setBannerPhoto(employeeId, fileId, versionNumber)
  Set an uploaded file as an employee’s banner photo.
- setBannerPhoto(employeeId, fileUpload)
  Set a file that hasn’t been uploaded as an employee’s banner photo.
- setBannerPhotoWithAttributes(employeeId, bannerPhoto)
  Set and crop an uploaded file as an employee’s banner photo.
- setBannerPhotoWithAttributes(employeeId, bannerPhoto, fileUpload)
  Set and crop a file that hasn’t been uploaded as an employee’s banner photo.
- setPhoto(employeeId, fileId, versionNumber)
  Set an uploaded file as an employee’s photo.
- setPhoto(employeeId, fileUpload)
  Set a file that hasn’t been uploaded as an employee’s photo.
- setPhotoWithAttributes(employeeId, photo)
  Set and crop an uploaded file as an employee’s photo.
- setPhotoWithAttributes(employeeId, photo, fileUpload)
  Set and crop a file that hasn’t been uploaded as an employee’s photo.

**deleteBannerPhoto(employeeId)**

Delete an employee’s banner photo.

**API Version**

51.0

**Requires Chatter**

No
Signature

```java
public static Void deleteBannerPhoto(String employeeId)
```

Parameters

- `employeeId`
  - Type: `String`
  - ID of the employee.

Return Value

Type: `Void`

```
deletePhoto(employeeId)
```
Delete an employee’s photo.

API Version

51.0

Requires Chatter

No

Signature

```java
public static Void deletePhoto(String employeeId)
```

Parameters

- `employeeId`
  - Type: `String`
  - ID of the employee.

Return Value

Type: `Void`

```
getBannerPhoto(employeeId)
```
Get an employee’s banner photo.

API Version

51.0

Requires Chatter

No
Signature

```java
public static ConnectApi.BannerPhoto getBannerPhoto(String employeeId)
```

Parameters

- `employeeId`
  - Type: `String`
  - ID of the employee.

Return Value

Type: `ConnectApi.BannerPhoto`

```java
getPhoto(employeeId)
```

Get an employee’s photo.

API Version

51.0

Available to Guest Users

51.0

Requires Chatter

No

Signature

```java
public static ConnectApi.Photo getPhoto(String employeeId)
```

Parameters

- `employeeId`
  - Type: `String`
  - ID of the employee.

Return Value

Type: `ConnectApi.Photo`

```java
setBannerPhoto(employeeId, fileId, versionNumber)
```

Set an uploaded file as an employee’s banner photo.

API Version

51.0
Requires Chatter
No

Signature
public static ConnectApi.BannerPhoto setBannerPhoto(String employeeId, String fileId, Integer versionNumber)

Parameters
employeeId
Type: String
ID of the employee.

fileId
Type: String
ID of the uploaded file to use as the employee banner photo. The file must be an image and be smaller than 2 GB.

versionNumber
Type: Integer
Version number of the file. Specify an existing version number or, to get the latest version, specify null.

Return Value
Type: ConnectApi.BannerPhoto

setBannerPhoto(employeeId, fileUpload)
Set a file that hasn't been uploaded as an employee's banner photo.

API Version
51.0

Requires Chatter
No

Signature
public static ConnectApi.BannerPhoto setBannerPhoto(String employeeId, ConnectApi.BinaryInput fileUpload)

Parameters
employeeId
Type: String
ID of the employee.

fileUpload
Type: ConnectApi.BinaryInput
File to use as the photo. The content type must be usable as an image.

Return Value
Type: ConnectApi.BannerPhoto

Usage
Photos are processed asynchronously and might not be visible right away.

setBannerPhotoWithAttributes(employeeId, bannerPhoto)
Set and crop an uploaded file as an employee's banner photo.

API Version
51.0

Requires Chatter
No

Signature
public static ConnectApi.BannerPhoto setBannerPhotoWithAttributes(String employeeId, ConnectApi.BannerPhotoInput bannerPhoto)

Parameters
employeeId
  Type: String
  ID of the employee.

bannerPhoto
  Type: ConnectApi.BannerPhotoInput
  A ConnectApi.BannerPhotoInput object that specifies the ID and version of the file, and how to crop the file.

Return Value
Type: ConnectApi.BannerPhoto

Usage
Photos are processed asynchronously and might not be visible right away.

setBannerPhotoWithAttributes(employeeId, bannerPhoto, fileUpload)
Set and crop a file that hasn't been uploaded as an employee's banner photo.
**API Version**

$1.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.BannerPhoto setBannerPhotoWithAttributes(String employeeId,
    ConnectApi.BannerPhotoInput bannerPhoto, ConnectApi.BinaryInput fileUpload)
```

**Parameters**

- `employeeId`
  - Type: `String`
  - ID of the employee.

- `bannerPhoto`
  - Type: `ConnectApi.BannerPhotoInput`
  - A `ConnectApi.BannerPhotoInput` object specifying the cropping parameters.

- `fileUpload`
  - Type: `ConnectApi.BinaryInput`
  - File to use as the photo. The content type must be usable as an image.

**Return Value**

Type: `ConnectApi.BannerPhoto`

**Usage**

Photos are processed asynchronously and might not be visible right away.

**setPhoto(employeeId, fileId, versionNumber)**

Set an uploaded file as an employee’s photo.

**API Version**

$1.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.Photo setPhoto(String employeeId, String fileId, Integer
    versionNumber)
```
Parameters

employeeId
Type: String
ID of the employee.

fileId
Type: String
ID of the uploaded file to use as the employee photo. The file must be an image and be smaller than 2 GB.

versionNumber
Type: Integer
Version number of the file. Specify an existing version number or, to get the latest version, specify null.

Return Value
Type: ConnectApi.Photo

Usage
Photos are processed asynchronously and might not be visible right away.

setPhoto(employeeId, fileId)
Set a file that hasn’t been uploaded as an employee’s photo.

API Version
51.0

Requires Chatter
No

Signature
public static ConnectApi.Photo setPhoto(String employeeId, ConnectApi.BinaryInput fileUpload)

Parameters

employeeId
Type: String
ID of the employee.

fileUpload
Type: ConnectApi.BinaryInput
File to use as the photo. The content type must be usable as an image.
Return Value
Type: ConnectApi.Photo

Usage
Photos are processed asynchronously and might not be visible right away.

**setPhotoWithAttributes(employeeId, photo)**
Set and crop an uploaded file as an employee's photo.

API Version
51.0

Requires Chatter
No

Signature
```
public static ConnectApi.Photo setPhotoWithAttributes(String employeeId,
ConnectApi.PhotoInput photo)
```

Parameters
- **employeeId**
  Type: String
  ID of the employee.
- **photo**
  Type: ConnectApi.PhotoInput
  A ConnectApi.PhotoInput object specifying the file ID, version number, and cropping parameters.

Return Value
Type: ConnectApi.Photo

Usage
Photos are processed asynchronously and might not be visible right away.

**setPhotoWithAttributes(employeeId, photo, fileUpload)**
Set and crop a file that hasn't been uploaded as an employee's photo.

API Version
51.0
Requires Chatter
No

Signature
public static ConnectApi.Photo setPhotoWithAttributes(String employeeId, ConnectApi.PhotoInput photo, ConnectApi.BinaryInput fileUpload)

Parameters
employeeId
Type: String
ID of the employee.

photo
Type: ConnectApi.PhotoInput
A ConnectApi.PhotoInput object specifying the cropping parameters.

fileUpload
Type: ConnectApi.BinaryInput
File to use as the photo. The content type must be usable as an image.

Return Value
Type: ConnectApi.Photo

Usage
Photos are processed asynchronously and might not be visible right away.

ExternalEmailServices Class
Access information about integration with external email services, such as sending email within Salesforce through an external email account.

Namespace
ConnectApi

External Email Services Methods
The following are methods for ExternalEmailService. All methods are static.

IN THIS SECTION:
  getUserOauthInfo(landingPage)
  Get information about whether an external email service has been authorized to send email on behalf of a user.
**getUserOauthInfo (landingPage)**
Get information about whether an external email service has been authorized to send email on behalf of a user.

**API Version**
37.0

**Requires Chatter**
No

**Signature**
```java
public static getUserOauthInfo(String landingPage)
```

**Parameters**
- `landingPage`
  - **Type:** String
  - The landing page that the user starts on when they are finished with the OAuth authorization process.

**Return Value**
Type: `ConnectApi.UserOauthInfo`

**SEE ALSO:**
- *Apex Developer Guide: Testing ConnectApi Code*

---

**ExternalManagedAccount Class**
Get externally managed accounts.

**Namespace**
`ConnectApi`

**ExternalManagedAccount Methods**
These methods are for `ExternalManagedAccount`. All methods are static.

**IN THIS SECTION:**
- `getCommunitiesExternalManagedAccounts(communityId)`
  Get externally managed accounts available to the context user across all Experience Cloud sites.
- `getCommunitiesExternalManagedAccounts(communityId, includeMyAccount)`
  Get externally managed accounts available to the context user, including the context user’s account, across all Experience Cloud sites.
getExternalManagedAccounts(webstoreId)
Get externally managed accounts for a store.

ggetExternalManagedAccounts(webstoreId, includeMyAccount)
Get externally managed accounts, including the context user’s account, for a store.

gCommunitiesExternalManagedAccounts (communityId)
Get externally managed accounts available to the context user across all Experience Cloud sites.

API Version
50.0

Requires Chatter
No

Signature

public static ConnectApi.ExternalManagedAccountCollectionOutput getCommunitiesExternalManagedAccounts(String communityId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

Note: Regardless of the ID specified, this method returns externally managed accounts available to the context user across all Experience Cloud sites.

Return Value
Type: ConnectApi.ExternalManagedAccountCollectionOutput

gCommunitiesExternalManagedAccounts (communityId, includeMyAccount)
Get externally managed accounts available to the context user, including the context user’s account, across all Experience Cloud sites.

API Version
53.0

Requires Chatter
No
public static ConnectApi.ExternalManagedAccountCollectionOutput getCommunitiesExternalManagedAccounts(String communityId, Boolean includeMyAccount)

Parameters

`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

ℹ️ **Note:** Regardless of the ID specified, this method returns externally managed accounts available to the context user across all Experience Cloud sites.

`includeMyAccount`
Type: `Boolean`
Specifies whether to return the context user’s account (`true`) or not (`false`). The default value is `false`.

Return Value
Type: `ConnectApi.ExternalManagedAccountCollectionOutput`

```java
getExternalManagedAccounts(String webstoreId)
```
Get externally managed accounts for a store.

API Version
49.0

Requires Chatter
No

Signature
public static ConnectApi.ExternalManagedAccountCollectionOutput getExternalManagedAccounts(String webstoreId)

Parameters

`webstoreId`
Type: `String`
ID of the webstore.

Return Value
Type: `ConnectApi.ExternalManagedAccountCollectionOutput`
getExternalManagedAccounts (webstoreId, includeMyAccount)
Get externally managed accounts, including the context user's account, for a store.

API Version
53.0

Requires Chatter
No

Signature
public static ConnectApi.ExternalManagedAccountCollectionOutput getExternalManagedAccounts(String webstoreId, Boolean includeMyAccount)

Parameters
webstoreId
Type: String
ID of the webstore.

includeMyAccount
Type: Boolean
Specifies whether to return the context user's account (true) or not (false). The default value is false.

Return Value
Type: ConnectApi.ExternalManagedAccountCollectionOutput

FieldService Class
Preview and create shifts from a pattern or filter fields on recordset filter criteria.

Namespace
ConnectApi

FieldService Methods
These methods are for FieldService. All methods are static.

IN THIS SECTION:
createShiftsFromPattern(shiftsFromPatternInput, shiftPatternId)
Create up to 2,000 shifts from a pattern.

evaluateRecordsetFilterCriteria(recordsetFilterCriteriaInput)
Filter records on recordset filter criteria.
previewShiftsFromPattern(shiftsFromPatternInput, shiftPatternId)
Preview up to 2,000 shifts from a pattern.

createShiftsFromPattern(shiftsFromPatternInput, shiftPatternId)
Create up to 2,000 shifts from a pattern.

API Version
S1.0

Requires Chatter
Yes

Signature
public static ConnectApi.ShiftsFromPattern createShiftsFromPattern(ConnectApi.ShiftsFromPatternInput shiftsFromPatternInput, String shiftPatternId)

Parameters
shiftsFromPatternInput
Type: ConnectApi.ShiftsFromPatternInput
A ConnectApi.ShiftsFromPatternInput object providing the pattern.

shiftPatternId
Type: String
ID of the shift pattern.

Return Value
Type: ConnectApi.ShiftsFromPattern

evaluateRecordsetFilterCriteria(recordsetFilterCriteriaInput)
Filter records on recordset filter criteria.

API Version
S3.0

Requires Chatter
No
Signature

```java
public static ConnectApi.RecordsetFilterCriteriaOutput
evaluateRecordsetFilterCriteria(ConnectApi.RecordsetFilterCriteriaInput
recordsetFilterCriteriaInput)
```

Parameters

```java
recordsetFilterCriteriaInput
Type: ConnectApi.RecordsetFilterCriteriaInput
An ConnectApi.RecordsetFilterCriteriaInput object providing a set of recordset filter criteria and records.
```

Return Value

```java
Type: ConnectApi.RecordsetFilterCriteriaOutput
```

Usage

Field service must be enabled.

```java
previewShiftsFromPattern(shiftsFromPatternInput, shiftPatternId)
```

Preview up to 2,000 shifts from a pattern.

API Version

51.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.ShiftsFromPattern
previewShiftsFromPattern(ConnectApi.ShiftsFromPatternInput shiftsFromPatternInput,
String shiftPatternId)
```

Parameters

```java
shiftsFromPatternInput
Type: ConnectApi.ShiftsFromPatternInput
A ConnectApi.ShiftsFromPatternInput object providing the pattern.

shiftPatternId
Type: String
ID of the shift pattern.
```

Return Value

```java
Type: ConnectApi.ShiftsFromPattern
```
FulfillmentOrder Class

Fulfill orders in Order Management.

Namespace

ConnectApi

FulfillmentOrder Methods

The following are methods for FulfillmentOrder. All methods are static.

IN THIS SECTION:

- cancelFulfillmentOrderLineItems(fulfillmentOrderId, cancelFulfillmentOrderLineItemsInput)
  Cancel FulfillmentOrderLineItems from a FulfillmentOrder. This action doesn’t cancel the associated OrderItemSummaries, so reallocate the canceled quantities to a new FulfillmentOrder.

- createFulfillmentOrders(fulfillmentOrderInput)
  Create one or more FulfillmentOrders and FulfillmentOrderLineItems for an OrderDeliveryGroupSummary, which defines a delivery method and recipient for an OrderSummary. You specify the OrderItemSummaries to allocate, which can be fulfilled from different locations. Specifying multiple fulfillment groups creates one FulfillmentOrder for each location. For each OrderItemSummary, a FulfillmentOrderLineItem is created and assigned to the corresponding FulfillmentOrder.

- createInvoice(fulfillmentOrderId, invoiceInput)
  Create an invoice for a FulfillmentOrder that doesn’t have one.

- createMultipleFulfillmentOrder(multipleFulfillmentOrderInput)
  Create FulfillmentOrders for multiple OrderDeliveryGroups in a single request.

- createMultipleInvoices(invoicesInput)
  Create Invoices for multiple FulfillmentOrders.

#### cancelFulfillmentOrderLineItems (fulfillmentOrderId, cancelFulfillmentOrderLineItemsInput)

Cancel FulfillmentOrderLineItems from a FulfillmentOrder. This action doesn’t cancel the associated OrderItemSummaries, so reallocate the canceled quantities to a new FulfillmentOrder.

**API Version**

48.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.FulfillmentOrderCancelLineItemsOutputRepresentation cancelFulfillmentOrderLineItems(String fulfillmentOrderId,
```

1241
ConnectApi.FulfillmentOrderLineItemsToCancelInputRepresentation
cancelFulfillmentOrderLineItemsInput)

Parameters

fulfillmentOrderId
  Type: String
  ID of the FulfillmentOrder.

cancelFulfillmentOrderLineItemsInput
  Type: ConnectApi.FulfillmentOrderLineItemsToCancelInputRepresentation
  List of FulfillmentOrderLineItems to cancel.

Return Value

Type: ConnectApi.FulfillmentOrderCancelLineItemsOutputRepresentation

Example

```java
String fulfillmentOrderId = '0a3xx0000000085AAA';
List<ConnectApi.FulfillmentOrderLineItemInputRepresentation> itemToCancelList = new
List<ConnectApi.FulfillmentOrderLineItemInputRepresentation>);

for(FulfillmentOrderLineItem fulfillmentOrderLineItem :
fulfillmentOrder.FulfillmentOrderLineItems)
  {
    ConnectApi.FulfillmentOrderLineItemInputRepresentation itemToCancel = new
    ConnectApi.FulfillmentOrderLineItemInputRepresentation();
    itemToCancel.fulfillmentOrderLineItemId = fulfillmentOrderLineItem.Id;
    itemToCancel.quantity = 1;
    itemToCancelList.add(itemToCancel);
  }

ConnectAPI.FulfillmentOrderLineItemsToCancelInputRepresentation input = new
ConnectAPI.FulfillmentOrderLineItemsToCancelInputRepresentation();
input.fulfillmentOrderLineItemsToCancel = itemToCancelList;

ConnectAPI.FulfillmentOrderCancelLineItemsOutputRepresentation result =
ConnectAPI.FulfillmentOrder.cancelFulfillmentOrderLineItems(fulfillmentOrderId, input);
```

createFulfillmentOrders(fulfillmentOrderInput)

Create one or more FulfillmentOrders and FulfillmentOrderLineItems for an OrderDeliveryGroupSummary, which defines a delivery method and recipient for an OrderSummary. You specify the OrderItemSummaries to allocate, which can be fulfilled from different locations. Specifying multiple fulfillment groups creates one FulfillmentOrder for each location. For each OrderItemSummary, a FulfillmentOrderLineItem is created and assigned to the corresponding FulfillmentOrder.

API Version

48.0
Requires Chatter
No

Signature

```java
public static ConnectApi.FulfillmentOrderOutputRepresentation createFulfillmentOrders(ConnectApi.FulfillmentOrderInputRepresentation fulfillmentOrderInput)
```

Parameters

`fulfillmentOrderInput`
- Type: `ConnectApi.FulfillmentOrderInputRepresentation`
  - OrderItemSummaries to allocate, with location and delivery information.

Return Value

Type: `ConnectApi.FulfillmentOrderOutputRepresentation`

Example

```java
String orderSummaryId = '1Osx0000004CCG';
String fulfillmentType = 'warehouse';
String warehouseFromLocationId = (SELECT Id from Location WHERE LocationType='Warehouse' LIMIT 1).Id;

ConnectApi.FulfillmentOrderInputRepresentation fulfillmentOrderInput = new ConnectApi.FulfillmentOrderInputRepresentation();
fulfillmentOrderInput.orderSummaryId = orderSummaryId;
List<OrderDeliveryGroupSummary> orderDeliveryGroupSummaryList = (SELECT Id FROM OrderDeliveryGroupSummary WHERE OrderSummaryId =: orderSummaryId);

for (OrderDeliveryGroupSummary orderDeliveryGroupSummary: orderDeliveryGroupSummaryList){

  fulfillmentOrderInput.orderDeliveryGroupSummaryId = orderDeliveryGroupSummary.Id;
  List<ConnectApi.FulfillmentGroupInputRepresentation> fulfillmentGroups = new List<ConnectApi.FulfillmentGroupInputRepresentation>();
  ConnectApi.FulfillmentGroupInputRepresentation fulfillmentGroup = new ConnectApi.FulfillmentGroupInputRepresentation();
  fulfillmentGroup.fulfilledFromLocationId = warehouseFromLocationId;
  fulfillmentGroup.fulfillmentType = fulfillmentType;
  fulfillmentGroups.add(fulfillmentGroup);

  List<ConnectApi.OrderItemSummaryInputRepresentation> orderItemSummaries = new List<ConnectApi.OrderItemSummaryInputRepresentation>();
  List<OrderItemSummary> orderItemSummaryList = (Select Id, quantity FROM OrderItemSummary WHERE OrderSummaryId =: orderSummaryId AND OrderDeliveryGroupSummaryId =: orderDeliveryGroupSummary.Id);
  for(OrderItemSummary orderItemSummary : orderItemSummaryList){
```
createInvoice(fulfillmentOrderId, invoiceInput)

Create an invoice for a FulfillmentOrder that doesn’t have one.

API Version
48.0

Requires Chatter
No

Signature
public static ConnectApi.FulfillmentOrderInvoiceOutputRepresentation createInvoice(String fulfillmentOrderId, ConnectApi.FulfillmentOrderInvoiceInputRepresentation invoiceInput)

Parameters
fulfillmentOrderId
Type: String
ID of the FulfillmentOrder.

invoiceInput
Type: ConnectApi.FulfillmentOrderInvoiceInputRepresentation
Required input with no data.

Return Value
Type: ConnectApi.FulfillmentOrderInvoiceOutputRepresentation

Example
String fulfillmentOrderId = '0a3xx0000000085AAA';
createMultipleFulfillmentOrder (multipleFulfillmentOrderInput)
Create FulfillmentOrders for multiple OrderDeliveryGroups in a single request.

API Version
50.0

Requires Chatter
No

Signature
public static ConnectApi.MultipleFulfillmentOrderOutputRepresentation createMultipleFulfillmentOrder(ConnectApi.MultipleFulfillmentOrderInputRepresentation multipleFulfillmentOrderInput)

Parameters
multipleFulfillmentOrderInput
Type: ConnectApi.MultipleFulfillmentOrderInputRepresentation
Wraps a list of inputs for creating fulfillment orders.

Return Value
Type: ConnectApi.MultipleFulfillmentOrderOutputRepresentation

createMultipleInvoices (invoicesInput)
Create Invoices for multiple FulfillmentOrders.

API Version
52.0

Requires Chatter
No

Signature
public static ConnectApi.MultipleFulfillmentOrderInvoicesOutputRepresentation createMultipleInvoices(ConnectApi.MultipleFulfillmentOrderInvoicesInputRepresentation invoicesInput)
Parameters

`invoicesInput`
Type: `ConnectApi.MultipleFulfillmentOrderInvoicesInputRepresentation`

The FulfillmentOrders to create Invoices for.

Return Value

Type: `ConnectApi.MultipleFulfillmentOrderInvoicesOutputRepresentation`

Knowledge Class

Get information about trending articles in Experience Cloud sites.

Namespace

`ConnectApi`

Knowledge Methods

The following are methods for `Knowledge`. All methods are static.

IN THIS SECTION:

- `getTopViewedArticlesForTopic(communityId, topicId, maxResults)`
  Get the top viewed articles for a topic.
- `getTrendingArticles(communityId, maxResults)`
  Get trending articles for an Experience Cloud site.
- `getTrendingArticlesForTopic(communityId, topicId, maxResults)`
  Get the trending articles for a topic in an Experience Cloud site.

`getTopViewedArticlesForTopic(communityId, topicId, maxResults)`
Get the top viewed articles for a topic.

API Version

41.0

Available to Guest Users

41.0

Requires Chatter

No
Signature

```java
public static ConnectApi.KnowledgeArticleVersionCollection getTopViewedArticlesForTopic(String communityId, String topicId, Integer maxResults)
```

Parameters

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **topicId**
  - Type: `String`
  - ID of the topic.

- **maxResults**
  - Type: `Integer`
  - The maximum number of articles returned for each topic ID. Values can be from 1 to 25. The default value is 5.

Return Value

Type: `ConnectApi.KnowledgeArticleVersionCollection`

```java
getTrendingArticles(communityId, maxResults)
```

Get trending articles for an Experience Cloud site.

API Version

36.0

Available to Guest Users

36.0

Requires Chatter

No

Signature

```java
public static ConnectApi.KnowledgeArticleVersionCollection getTrendingArticles(String communityId, Integer maxResults)
```

Parameters

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **maxResults**
  - Type: `Integer`
The maximum number of articles returned. Values can be from 0 to 25. Default is 5.

**Return Value**
Type: `ConnectApi.KnowledgeArticleVersionCollection`

**Usage**
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestGetTrendingArticles(communityId, maxResults, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

`getTrendingArticlesForTopic(communityId, topicId, maxResults)`
Get the trending articles for a topic in an Experience Cloud site.

**API Version**
36.0

**Available to Guest Users**
36.0

**Requires Chatter**
No

**Signature**
```
public static ConnectApi.KnowledgeArticleVersionCollection
getTrendingArticlesForTopic(String communityId, String topicId, Integer maxResults)
```

**Parameters**
- **communityId**
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.
- **topicId**
  Type: `String`
  ID of the topic.
- **maxResults**
  Type: `Integer`
  The maximum number of articles returned. Values can be from 0 to 25. Default is 5.
Return Value
Type: `ConnectApi.KnowledgeArticleVersionCollection`

Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestGetTrendingArticlesForTopic(communityId, topicId, maxResults, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

Knowledge Test Methods
The following are the test methods for `Knowledge`. All methods are static.

For information about using these methods to test your `ConnectApi` code, see *Testing ConnectApi Code*.

`setTestGetTrendingArticles(communityId, maxResults, result)`
Register a `ConnectApi.KnowledgeVersionArticleCollection` object to be returned when the matching `ConnectApi.getTrendingArticles` method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
36.0

Signature
```
public static Void setTestGetTrendingArticles(String communityId, Integer maxResults, ConnectApi.KnowledgeArticleVersionCollection result)
```

Parameters
- **communityId**
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.
- **maxResults**
  Type: `Integer`
  The maximum number of articles returned. Values can be from 0 to 25. Default is 5.
- **result**
  Type: `ConnectApi.KnowledgeArticleVersionCollection`
  Object containing test data.
setTestGetTrendingArticlesForTopic(communityId, topicId, maxResults, result)

Register a ConnectApi.KnowledgeVersionArticleCollection object to be returned when the matching ConnectApi.getTrendingArticlesForTopic method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
36.0

Signature
public static Void setTestGetTrendingArticlesForTopic(String communityId, String topicId, Integer maxResults, ConnectApi.KnowledgeArticleVersionCollection result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

topicId
Type: String
ID of the topic.

maxResults
Type: Integer
The maximum number of articles returned. Values can be from 0 to 25. Default is 5.

result
Type: ConnectApi.KnowledgeArticleVersionCollection
Object containing test data.

Return Value
Type: Void

SEE ALSO:
getTrendingArticles(communityId, maxResults)
LightningScheduler Class
Create and update service appointments.

Namespace
ConnectApi

LightningScheduler Methods
The following are methods for LightningScheduler. All methods are static.

IN THIS SECTION:
createServiceAppointment(createServiceAppointmentInput)
Create a service appointment.
updateServiceAppointment(updateServiceAppointmentInput)
Update a service appointment.

createServiceAppointment (createServiceAppointmentInput)
Create a service appointment.

API Version
53.0

Requires Chatter
No

Signature
public static ConnectApi.ServiceAppointmentOutput createServiceAppointment (ConnectApi.CreateServiceAppointmentInput createServiceAppointmentInput)

Parameters
createServiceAppointmentInput
   Type: ConnectApi.CreateServiceAppointmentInput
   Input parameters to create a service appointment.

Return Value
Type: ConnectApi.ServiceAppointmentOutput
Usage

Considerations for using engagement channel types with the service-appointments resource:

- Enable **Schedule Appointments Using Engagement Channels** in Salesforce Scheduler Settings in your Salesforce org.

- When you create or modify appointments, shifts must be defined in the scheduling policy. For more information on setting up shifts in the scheduling policy, see **Define Shift Rules in Scheduling Policy**.

  **Note:** Engagement channel types are not supported with operating hours rules in the scheduling policy.

- When you use engagement channel type and shifts to create a service appointment, Salesforce Scheduler considers the default value for the Appointment Type (if not specified). However, Salesforce Scheduler only considers the engagement channel type and Appointment Type is ignored.

Example

For an account (existing user):

```java
ConnectApi.ExtendedFieldInput extendedFieldEmail = new ConnectApi.ExtendedFieldInput();
extendedFieldEmail.name = 'Email';
extendedFieldEmail.value = 'rachael.adams@salesforce.com';

ConnectApi.ExtendedFieldInput extendedFieldPhone = new ConnectApi.ExtendedFieldInput();
extendedFieldPhone.name = 'Phone';
extendedFieldPhone.value = '1234567890';

List<ConnectApi.ExtendedFieldInput> extendedFieldList = new List<ConnectApi.ExtendedFieldInput>();
extendedFieldList.add(extendedFieldEmail);
extendedFieldList.add(extendedFieldPhone);

ConnectApi.ServiceAppointmentInput serviceAppInput = new ConnectApi.ServiceAppointmentInput();
serviceAppInput.extendedFields = extendedFieldList;
serviceAppInput.engagementChannelTypeId = '0eFRM00000000Bv2AI';
serviceAppInput.serviceTerritoryId = '0Hhx000000000004C92CAE';
serviceAppInput.workTypeId = '08qx000000000C92AAE';
serviceAppInput.parentRecordId = '001xx0000003GYR1AAO';
serviceAppInput.schedStartTime = DateTime.valueOf('2021-05-28 12:15:00');
serviceAppInput.schedEndTime = DateTime.valueOf('2021-05-28 12:45:00');

ConnectApi.AssignedResourcesInput asResourceInput = new ConnectApi.AssignedResourcesInput();
asResourceInput.serviceResourceId = '0Hnxx000000000004CAiCAM';
asResourceInput.isRequiredResource = true;
asResourceInput.isPrimaryResource = true;

List<ConnectApi.AssignedResourcesInput> asResourceInputList = new List<ConnectApi.AssignedResourcesInput>();
asResourceInputList.add(asResourceInput);

ConnectApi.CreateServiceAppointmentInput createInput = new ConnectApi.CreateServiceAppointmentInput();
createInput.serviceAppointment = serviceAppInput;
createInput.assignedResources = asResourceInputList;
```
try{
    ConnectApi.ServiceAppointmentOutput appointmentResult =
    ConnectApi.LightningScheduler.createServiceAppointment(createInput);
    String serviceAppointmentId = appointmentResult.result.serviceAppointmentId;
    List<String> assignedResourceIds = appointmentResult.result.assignedResourceIds;
}catch(ConnectApi.ConnectApiException ex){
    //Handle Exception
}

For a lead (authenticated guest user):

ConnectApi.LeadInput leadInput = new ConnectApi.LeadInput();
leadInput.firstName = 'Rachel';
leadInput.lastName = 'Adams';
leadInput.phone = '012-345-6789';
leadInput.email = 'rachel.adams@salesforce.com';
leadInput.company = 'Salesforce';

ConnectApi.ExtendedFieldInput extendedFieldEmail = new ConnectApi.ExtendedFieldInput();
extendedFieldEmail.name = 'Email';
extendedFieldEmail.value = 'rachael.adams@salesforce.com';

ConnectApi.ExtendedFieldInput extendedFieldPhone = new ConnectApi.ExtendedFieldInput();
extendedFieldPhone.name = 'Phone';
extendedFieldPhone.value = '1234567890';

List<ConnectApi.ExtendedFieldInput> extendedFieldList = new List<ConnectApi.ExtendedFieldInput>();
extendedFieldList.add(extendedFieldEmail);
extendedFieldList.add(extendedFieldPhone);

ConnectApi.ServiceAppointmentInput serviceAppInput = new ConnectApi.ServiceAppointmentInput();
serviceAppInput.extendedFields = extendedFieldList;
serviceAppInput.engagementChannelTypeId = '0eFRM00000000Bv2AI';
serviceAppInput.serviceTerritoryId = '0Hhxx0000004C92CAE';
serviceAppInput.workTypeId = '08qxx000000004C92AAE';
serviceAppInput.schedStartTime = DateTime.valueOf('2021-05-28 12:15:00');
serviceAppInput.schedEndTime = DateTime.valueOf('2021-05-28 12:45:00');

ConnectApi.AssignedResourcesInput asResourceInput = new ConnectApi.AssignedResourcesInput();
asResourceInput.serviceResourceId = '0Hnxx0000004CAiCAM';
asResourceInput.isRequiredResource = true;
asResourceInput.isPrimaryResource = true;

List<ConnectApi.AssignedResourcesInput> asResourceInputList = new List<ConnectApi.AssignedResourcesInput>();
asResourceInputList.add(asResourceInput);

ConnectApi.CreateServiceAppointmentInput createInput = new ConnectApi.CreateServiceAppointmentInput();
createInput.serviceAppointment = serviceAppInput;
createInput.assignedResources = asResourceInputList;
createInput.lead = leadInput;

try{
  ConnectApi.ServiceAppointmentOutput appointmentResult =
  ConnectApi.LightningScheduler.createServiceAppointment(createInput);
  String serviceAppointmentId = appointmentResult.result.serviceAppointmentId;
  List<String> assignedResourceIds = appointmentResult.result.assignedResourceIds;
}catch(ConnectApi.ConnectApiException ex){
  //Handle Exception
}

SEE ALSO:
  Service Appointments

**updateServiceAppointment(updateServiceAppointmentInput)**

Update a service appointment.

**API Version**

S3.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.ServiceAppointmentOutput
updateServiceAppointment(ConnectApi.UpdateServiceAppointmentInput
updateServiceAppointmentInput)
```

**Parameters**

`updateServiceAppointmentInput`  
Type: `ConnectApi.UpdateServiceAppointmentInput`  
Input parameters to update a service appointment.

**Return Value**

Type: `ConnectApi.ServiceAppointmentOutput`

**Usage**

Considerations for using engagement channel types with the `service-appointments` resource:

- Enable **Schedule Appointments Using Engagement Channels** in Salesforce Scheduler Settings in your Salesforce org.
- When you create or modify appointments, shifts must be defined in the scheduling policy. For more information on setting up shifts in the scheduling policy, see **Define Shift Rules in Scheduling Policy**.
Note: Engagement channel types are not supported with operating hours rules in the scheduling policy.

- When you use engagement channel type and shifts to modify an appointment, Salesforce Scheduler considers the default value for the Appointment Type (if not specified). However, Salesforce Scheduler only considers the engagement channel type and Appointment Type is ignored.

Example

```java
ConnectApi.ExtendedFieldInput extendedFieldEmail = new ConnectApi.ExtendedFieldInput();
extendedFieldEmail.name = 'Email';
extendedFieldEmail.value = 'rachel.adams@salesforce.com.example';

ConnectApi.ExtendedFieldInput extendedFieldPhone = new ConnectApi.ExtendedFieldInput();
extendedFieldPhone.name = 'Phone';
extendedFieldPhone.value = '0123456789';

ConnectApi.ExtendedFieldInput extendedFieldStatus = new ConnectApi.ExtendedFieldInput();
extendedFieldStatus.name = 'Status';
extendedFieldStatus.value = 'None';

List<ConnectApi.ExtendedFieldInput> extendedFieldList = new List<ConnectApi.ExtendedFieldInput>();
extendedFieldList.add(extendedFieldEmail);
extendedFieldList.add(extendedFieldPhone);
extendedFieldList.add(extendedFieldStatus);

ConnectApi.ServiceAppointmentInput serviceAppInput = new ConnectApi.ServiceAppointmentInput();
serviceAppInput.extendedFields = extendedFieldList;
serviceAppInput.serviceTerritoryId = '0Hhx00000004C92CAE';
serviceAppInput.workTypeId = '08qx00000004C92AAE';
serviceAppInput.schedStartTime = DateTime.valueOf('2021-05-28 12:15:00');
serviceAppInput.schedEndTime = DateTime.valueOf('2021-05-28 12:45:00');

ConnectApi.AssignedResourcesInput asResourceInput = new ConnectApi.AssignedResourcesInput();
asResourceInput.serviceResourceId = '0Hnx00000004CAiCAM';
asResourceInput.isRequiredResource = true;
asResourceInput.isPrimaryResource = true;

//Multi-resource
ConnectApi.AssignedResourcesInput asResourceInputReq = new ConnectApi.AssignedResourcesInput();
asResourceInputReq.serviceResourceId = '0Hnx00000004CAgCAM';
asResourceInputReq.isRequiredResource = true;
asResourceInputReq.isPrimaryResource = false;

List<ConnectApi.AssignedResourcesInput> asResourceInputList = new List<ConnectApi.AssignedResourcesInput>();
asResourceInputList.add(asResourceInput);
asResourceInputList.add(asResourceInputReq);

ConnectApi.UpdateServiceAppointmentInput updateInput = new ConnectApi.UpdateServiceAppointmentInput();
```
ConnectApi.UpdateServiceAppointmentInput();
updateInput.serviceAppointment = serviceAppInput;
updateInput.assignedResources = asResourceInputList;
updateInput.serviceAppointmentId = '08pxx0000004CYqAAM';

try{
    ConnectApi.ServiceAppointmentOutput appointmentResult =
    ConnectApi.LightningScheduler.updateServiceAppointment(updateInput);
    String serviceAppointmentId = appointmentResult.result.serviceAppointmentId;
    List<String> assignedResourceIds = appointmentResult.result.assignedResourceIds;
}catch(ConnectApi.ConnectApiException ex){
    //Handle Exception
}

SEE ALSO:
Service Appointments

**ManagedContent Class**
Get managed content versions. Get a managed content space.

**Namespace**
ConnectApi

**ManagedContent Methods**
The following are methods for ManagedContent. All methods are static.

IN THIS SECTION:
- getAllContent(channelId, pageParam, pageSize, language, managedContentType, includeMetadata, startDate, endDate)
  Get all managed content versions for a channel.
- getAllContent(channelId, pageParam, pageSize, language, managedContentType, includeMetadata, startDate, endDate, showAbsoluteUrl)
  Get all managed content versions for a channel with absolute URLs.
- getAllDeliveryChannels(pageParam, pageSize)
  Get managed content delivery channels for the context user.
- getAllManagedContent(communityId, pageParam, pageSize, language, managedContentType)
  Get all managed content versions for an Experience Cloud site.
- getAllManagedContent(communityId, pageParam, pageSize, language, managedContentType, showAbsoluteUrl)
  Get all managed content versions for an Experience Cloud site with absolute URLs.
- getContentByContentKeys(channelId, contentKeys, pageParam, pageSize, language, managedContentType, includeMetadata, startDate, endDate, showAbsoluteUrl)
  Get managed content versions for a channel using a list of content keys.
getManagedContentByIds(channelId, managedContentIds, pageParam, pageSize, language, managedContentType)
Get managed content versions for a channel using a list of managed content IDs.

calculateByldsWith(channelId, managedContentIds, pageParam, pageSize, language, managedContentType, includeMetadata, startDate, endDate, showAbsoluteUrl)
Get managed content versions for a channel with absolute URLs using a list of managed content IDs.

callManagedContentByContentKeys(communityId, contentKeys, pageParam, pageSize, language, managedContentType, showAbsoluteUrl)
Get managed content versions for an Experience Cloud site using a list of content keys.

callManagedContentByldsWith(communityId, managedContentIds, pageParam, pageSize, language, managedContentType)
Get managed content versions for an Experience Cloud site using a list of managed content IDs.

callManagedContentByldsWith(communityId, managedContentIds, pageParam, pageSize, language, managedContentType, showAbsoluteUrl)
Get managed content versions for an Experience Cloud site with absolute URLs using a list of managed content IDs.

callManagedContentByTopics(communityId, topics, pageParam, pageSize, language, managedContentType)
Get managed content versions using a list of content topic names.

callManagedContentByTopics(communityId, topics, pageParam, pageSize, language, managedContentType, showAbsoluteUrl)
Get managed content versions with absolute URLs using a list of content topic names.

callManagedContentByTopicsAndContentKeys(communityId, contentKeys, topics, pageParam, pageSize, language, managedContentType, showAbsoluteUrl)
Get managed content versions using a list of content keys and content topic names.

callManagedContentByTopicsAndIds(communityId, managedContentIds, topics, pageParam, pageSize, language, managedContentType)
Get managed content versions using a list of managed content IDs and content topic names.

callManagedContentByTopicsAndIds(communityId, managedContentIds, topics, pageParam, pageSize, language, managedContentType, showAbsoluteUrl)
Get managed content versions with absolute URLs using a list of managed content IDs and content topic names.

callManagedContentSpace(contentSpaceId)
Get a managed content space.

callAllContent(channelId, pageParam, pageSize, language, managedContentType, includeMetadata, startDate, endDate)
Get all managed content versions for a channel.

**API Version**
48.0

**Available to Guest Users**
48.0
Requires Chatter

No

Signature

```java
public static ConnectApi.ManagedContentVersionCollection getAllContent(String channelId,
Integer pageParam, Integer pageSize, String language, String managedContentType, Boolean
includeMetadata, String startDate, String endDate)
```

Parameters

- **channelId**
  Type: String
  ID of the channel.

- **pageParam**
  Type: Integer
  Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

- **pageSize**
  Type: Integer
  Specifies the number of items per page. Valid values are from 1 through 250. For performance reasons, we recommend 25 or fewer items per page. If you pass in `null`, the default size is 25.

- **language**
  Type: String
  Language locale for the managed content, for example, `en_US`. If the requested translation isn't available, the language defaults to the context user's language. If the context user's language isn't available, the language defaults to the content type's original language.

- **managedContentType**
  Type: String
  Developer name of the content type, such as `cms_document` or `cms_image`.

- **includeMetadata**
  Type: Boolean
  Specifies whether to include metadata in the response (`true`) or not (`false`). The default value is `false`.

- **startDate**
  Type: String
  Publish start date in ISO 8601 format, for example, `2011-02-25T18:24:31.000Z`.

- **endDate**
  Type: String
  Publish end date in ISO 8601 format, for example, `2011-02-25T18:24:31.000Z`.

Return Value

Type: `ConnectApi.ManagedContentVersionCollection`
getAllContent(channelId, pageParam, pageSize, language, managedContentType, includeMetadata, startDate, endDate, showAbsoluteUrl)

Get all managed content versions for a channel with absolute URLs.

API Version
50.0

Available to Guest Users
50.0

Requires Chatter
No

Signature
public static ConnectApi.ManagedContentVersionCollection getAllContent(String channelId, Integer pageParam, Integer pageSize, String language, String managedContentType, Boolean includeMetadata, String startDate, String endDate, Boolean showAbsoluteUrl)

Parameters

channelId
  Type: String
  ID of the channel.

pageParam
  Type: Integer
  Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
  Type: Integer
  Specifies the number of items per page. Valid values are from 1 through 250. For performance reasons, we recommend 25 or fewer items per page. If you pass in null, the default size is 25.

language
  Type: String
  Language locale for the managed content, for example, en_US. If the requested translation isn't available, the language defaults to the context user's language. If the context user's language isn't available, the language defaults to the content type's original language.

managedContentType
  Type: String
  Developer name of the content type, such as cms_document or cms_image.

includeMetadata
  Type: Boolean
  Specifies whether to include metadata in the response (true) or not (false). The default value is false.
**startDate**

Type: String

Publish start date in ISO 8601 format, for example, 2011-02-25T18:24:31.000Z.

**endDate**

Type: String

Publish end date in ISO 8601 format, for example, 2011-02-25T18:24:31.000Z.

**showAbsoluteUrl**

Type: Boolean

Specifies whether to show absolute URLs in the output class (true) or not (false). The default value is false.

**Return Value**

Type: `ConnectApi.ManagedContentVersionCollection`

**getAllDeliveryChannels(pageParam, pageSize)**

Get managed content delivery channels for the context user.

**API Version**

48.0

**Available to Guest Users**

48.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.ManagedContentChannelCollection getAllDeliveryChannels(Integer pageParam, Integer pageSize)
```

**Parameters**

- **pageParam**
  
  Type: `Integer`

  Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

- **pageSize**
  
  Type: `Integer`

  Specifies the number of items per page. Valid values are from 1 through 250. If you pass in null, the default size is 25.

**Return Value**

Type: `ConnectApi.ManagedContentChannelCollection`
getAllManagedContent(communityId, pageParam, pageSize, language, managedContentType)

Get all managed content versions for an Experience Cloud site.

API Version
47.0

Available to Guest Users
47.0

Requires Chatter
No

Signature
public static ConnectApi.ManagedContentVersionCollection getAllManagedContent(String communityId, Integer pageParam, Integer pageSize, String language, String managedContentType)

Parameters

communityId
Type: String
ID of the Experience Cloud site.

pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 250. For performance reasons, we recommend 25 or fewer items per page. If you pass in null, the default size is 25.

language
Type: String
Language locale for the managed content, for example, en_US. If the requested translation isn’t available, the language defaults to the context user’s language. If the context user’s language isn’t available, the language defaults to the content type’s original language.

managedContentType
Type: String
Developer name of the content type, such as cms_document or cms_image.

Return Value
Type: ConnectApi.ManagedContentVersionCollection
getAllManagedContent(communityId, pageParam, pageSize, language, managedContentType, showAbsoluteUrl)

Get all managed content versions for an Experience Cloud site with absolute URLs.

API Version
50.0

Available to Guest Users
50.0

Requires Chatter
No

Signature
public static ConnectApi.ManagedContentVersionCollection getAllManagedContent(String communityId, Integer pageParam, Integer pageSize, String language, String managedContentType, Boolean showAbsoluteUrl)

Parameters

communityId
  Type: String
  ID of the Experience Cloud site.

pageParam
  Type: Integer
  Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
  Type: Integer
  Specifies the number of items per page. Valid values are from 1 through 250. For performance reasons, we recommend 25 or fewer items per page. If you pass in null, the default size is 25.

language
  Type: String
  Language locale for the managed content, for example, en_US. If the requested translation isn’t available, the language defaults to the context user’s language. If the context user’s language isn’t available, the language defaults to the content type’s original language.

managedContentType
  Type: String
  Developer name of the content type, such as cms_document or cms_image.

showAbsoluteUrl
  Type: Boolean
  Specifies whether to show absolute URLs in the output class (true) or not (false). The default value is false.
Return Value
Type: `ConnectApi.ManagedContentVersionCollection`

`getContentByContentKeys(channelId, contentKeys, pageParam, pageSize, language, managedContentType, includeMetadata, startDate, endDate, showAbsoluteUrl)`
Get managed content versions for a channel using a list of content keys.

API Version
51.0

Available to Guest Users
51.0

Requires Chatter
No

Signature
```
public static ConnectApi.ManagedContentVersionCollection getContentByContentKeys(String
channelId, List<String> contentKeys, Integer pageParam, Integer pageSize, String
language, String managedContentType, Boolean includeMetadata, String startDate, String
date, Boolean showAbsoluteUrl)
```

Parameters
```
channelId
  Type: String
  ID of the channel.

contentKeys
  Type: List<String>
  List of up to 50 content keys for the managed content. A content key is a universally unique identifier (UUID) such as MCA4CV5QS28ABSH7YRCPWCWGZQ.

pageParam
  Type: Integer
  Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
  Type: Integer
  Specifies the number of items per page. Valid values are from 1 through 250. For performance reasons, we recommend 25 or fewer items per page. If you pass in null, the default size is 25.

language
  Type: String
Language locale for the managed content, for example, en_US. If the requested translation isn’t available, the language defaults to the context user’s language. If the context user’s language isn’t available, the language defaults to the content type’s original language.

managedContentType
Type: String
Developer name of the content type, such as cms_document or cms_image.

includeMetadata
Type: Boolean
Specifies whether to include metadata in the response (true) or not (false). The default value is false.

startDate
Type: String
Publish start date in ISO 8601 format, for example, 2011-02-25T18:24:31.000Z.

date
Type: String
Publish end date in ISO 8601 format, for example, 2011-02-25T18:24:31.000Z.

showAbsoluteUrl
Type: Boolean
Specifies whether to show absolute URLs in the output class (true) or not (false). The default value is false.

Return Value
Type: ConnectApi.ManagedContentVersionCollection

getContentByIds(channelId, managedContentIds, pageParam, pageSize, language, managedContentType, includeMetadata, startDate, endDate)
Get managed content versions for a channel using a list of managed content IDs.

API Version
48.0

Available to Guest Users
48.0

Requires Chatter
No

Signature
public static ConnectApi.ManagedContentVersionCollection getContentByIds(String
channelId, List<String> managedContentIds, Integer pageParam, Integer pageSize, String
language, String managedContentType, Boolean includeMetadata, String startDate, String
endDate)
Parameters

`channelId`
Type: `String`
ID of the channel.

`managedContentIds`
Type: `List<String>`
Comma-separated list of managed content IDs.

`pageParam`
Type: `Integer`
Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

`pageSize`
Type: `Integer`
Specifies the number of items per page. Valid values are from 1 through 250. For performance reasons, we recommend 25 or fewer items per page. If you pass in `null`, the default size is 25.

`language`
Type: `String`
Language locale for the managed content, for example, `en_US`. If the requested translation isn’t available, the language defaults to the context user’s language. If the context user’s language isn’t available, the language defaults to the content type’s original language.

`managedContentType`
Type: `String`
Developer name of the content type, such as `cms_document` or `cms_image`.

`includeMetadata`
Type: `Boolean`
Specifies whether to include metadata in the response (`true`) or not (`false`). The default value is `false`.

`startDate`
Type: `String`
Publish start date in ISO 8601 format, for example, `2011-02-25T18:24:31.000Z`.

`endDate`
Type: `String`
Publish end date in ISO 8601 format, for example, `2011-02-25T18:24:31.000Z`.

Return Value

Type: `ConnectApi.ManagedContentVersionCollection`

`getContentByIds(channelId, managedContentIds, pageParam, pageSize, language, managedContentType, includeMetadata, startDate, endDate, showAbsoluteUrl)`

Get managed content versions for a channel with absolute URLs using a list of managed content IDs.
API Version
50.0

Available to Guest Users
50.0

Requires Chatter
No

Signature
```
public static ConnectApi.ManagedContentVersionCollection getContentByIds(String channelId, List<String> managedContentIds, Integer pageParam, Integer pageSize, String language, String managedContentType, Boolean includeMetadata, String startDate, String endDate, Boolean showAbsoluteUrl)
```

Parameters

**channelId**
Type: String
ID of the channel.

**managedContentIds**
Type: List<String>
Comma-separated list of managed content IDs.

**pageParam**
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

**pageSize**
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 250. For performance reasons, we recommend 25 or fewer items per page. If you pass in `null`, the default size is 25.

**language**
Type: String
Language locale for the managed content, for example, en_US. If the requested translation isn’t available, the language defaults to the context user’s language. If the context user’s language isn’t available, the language defaults to the content type’s original language.

**managedContentType**
Type: String
Developer name of the content type, such as cms_document or cms_image.

**includeMetadata**
Type: Boolean
Specifies whether to include metadata in the response (`true`) or not (`false`). The default value is `false`. 
**startDate**
Type: `String`
Publish start date in ISO 8601 format, for example, 2011-02-25T18:24:31.000Z.

**endDate**
Type: `String`
Publish end date in ISO 8601 format, for example, 2011-02-25T18:24:31.000Z.

**showAbsoluteUrl**
Type: `Boolean`
Specifies whether to show absolute URLs in the output class (true) or not (false). The default value is false.

**Return Value**
Type: `ConnectApi.ManagedContentVersionCollection`

**getManagedContentByContentKeys(communityId, contentKeys, pageParam, pageSize, language, managedContentType, showAbsoluteUrl)**
Get managed content versions for an Experience Cloud site using a list of content keys.

**API Version**
51.0

**Available to Guest Users**
51.0

**Requires Chatter**
No

**Signature**

```java
public static ConnectApi.ManagedContentVersionCollection
getManagedContentByContentKeys(String communityId, List<String> contentKeys, Integer pageParam, Integer pageSize, String language, String managedContentType, Boolean showAbsoluteUrl)
```

**Parameters**

**communityId**
Type: `String`
ID of the Experience Cloud site.

**contentKeys**
Type: `List<String>`
List of up to 50 content keys for the managed content. A content key is a universally unique identifier (UUID) such as MCA4CCV5QS2BAB5H7YRCRPTCWGZQ.
getPageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in \texttt{null} or 0, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 250. For performance reasons, we recommend 25 or fewer items per page. If you pass in \texttt{null}, the default size is 25.

language
Type: String
Language locale for the managed content, for example, \texttt{en-US}. If the requested translation isn't available, the language defaults to the context user's language. If the context user's language isn't available, the language defaults to the content type's original language.

managedContentType
Type: String
Developer name of the content type, such as \texttt{cms_document} or \texttt{cms_image}.

showAbsoluteUrl
Type: Boolean
Specifies whether to show absolute URLs in the output class (\texttt{true}) or not (\texttt{false}). The default value is \texttt{false}.

Return Value
Type: \texttt{ConnectApi.ManagedContentVersionCollection}

getManagedContentByIds(communityId, managedContentIds, pageParam, pageSize, language, managedContentType)
Get managed content versions for an Experience Cloud site using a list of managed content IDs.

API Version
47.0

Available to Guest Users
47.0

Requires Chatter
No

Signature
\texttt{public static ConnectApi.ManagedContentVersionCollection getManagedContentByIds(String communityId, List<String> managedContentIds, Integer pageParam, Integer pageSize, String language, String managedContentType)}
Parameters

- **communityId**
  - Type: `String`
  - ID of the Experience Cloud site.

- **managedContentIds**
  - Type: `List<String>`
  - Comma-separated list of managed content IDs.

- **pageParam**
  - Type: `Integer`
  - Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

- **pageSize**
  - Type: `Integer`
  - Specifies the number of items per page. Valid values are from 1 through 250. For performance reasons, we recommend 25 or fewer items per page. If you pass in `null`, the default size is 25.

- **language**
  - Type: `String`
  - Language locale for the managed content, for example, `en_US`. If the requested translation isn’t available, the language defaults to the context user’s language. If the context user’s language isn’t available, the language defaults to the content type’s original language.

- **managedContentType**
  - Type: `String`
  - Developer name of the content type, such as `cms_document` or `cms_image`.

Return Value

Type: `ConnectApi.ManagedContentVersionCollection`

`getManagedContentByIds(communityId, managedContentIds, pageParam, pageSize, language, managedContentType, showAbsoluteUrl)`

Get managed content versions for an Experience Cloud site with absolute URLs using a list of managed content IDs.

API Version

50.0

Available to Guest Users

50.0

Requires Chatter

No
Signature

```java
public static ConnectApi.ManagedContentVersionCollection getManagedContentByIds(String communityId, List<String> managedContentIds, Integer pageParam, Integer pageSize, String language, String managedContentType, Boolean showAbsoluteUrl)
```

Parameters

- **communityId**
  - Type: String
  - ID of the Experience Cloud site.

- **managedContentIds**
  - Type: List<String>
  - Comma-separated list of managed content IDs.

- **pageParam**
  - Type: Integer
  - Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

- **pageSize**
  - Type: Integer
  - Specifies the number of items per page. Valid values are from 1 through 250. For performance reasons, we recommend 25 or fewer items per page. If you pass in `null`, the default size is 25.

- **language**
  - Type: String
  - Language locale for the managed content, for example, `en_US`. If the requested translation isn’t available, the language defaults to the context user’s language. If the context user’s language isn’t available, the language defaults to the content type’s original language.

- **managedContentType**
  - Type: String
  - Developer name of the content type, such as `cms_document` or `cms_image`.

- **showAbsoluteUrl**
  - Type: Boolean
  - Specifies whether to show absolute URLs in the output class (`true`) or not (`false`). The default value is `false`.

Return Value

Type: `ConnectApi.ManagedContentVersionCollection`

```java
getManagedContentByTopics(communityId, topics, pageParam, pageSize, language, managedContentType)
```

Get managed content versions using a list of content topic names.

API Version

47.0
Available to Guest Users
47.0

Requires Chatter
No

Signature
public static ConnectApi.ManagedContentVersionCollection getManagedContentByTopics(String communityId, List<String> topics, Integer pageParam, Integer pageSize, String language, String managedContentType)

Parameters

communityId
Type: String
ID of the Experience Cloud site.

topics
Type: List<String>
Comma-separated list of up to 15 content topic names.

pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 250. For performance reasons, we recommend 25 or fewer items per page. If you pass in null, the default size is 25.

language
Type: String
Language locale for the managed content, for example, en_US. If the requested translation isn’t available, the language defaults to the context user’s language. If the context user’s language isn’t available, the language defaults to the content type’s original language.

managedContentType
Type: String
Developer name of the content type, such as cms_document or cms_image.

Return Value
Type: ConnectApi.ManagedContentVersionCollection

getManagedContentByTopics(communityId, topics, pageParam, pageSize, language, managedContentType, showAbsoluteUrl)
Get managed content versions with absolute URLs using a list of content topic names.
API Version
50.0

Available to Guest Users
50.0

Requires Chatter
No

Signature
public static ConnectApi.ManagedContentVersionCollection getManagedContentByTopics(String communityId, List<String> topics, Integer pageParam, Integer pageSize, String language, String managedContentType, Boolean showAbsoluteUrl)

Parameters
communityId
  Type: String
  ID of the Experience Cloud site.

topics
  Type: List<String>
  Comma-separated list of up to 15 content topic names.

pageParam
  Type: Integer
  Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
  Type: Integer
  Specifies the number of items per page. Valid values are from 1 through 250. For performance reasons, we recommend 25 or fewer items per page. If you pass in null, the default size is 25.

language
  Type: String
  Language locale for the managed content, for example, en_US. If the requested translation isn’t available, the language defaults to the context user’s language. If the context user’s language isn’t available, the language defaults to the content type’s original language.

managedContentType
  Type: String
  Developer name of the content type, such as cms_document or cms_image.

showAbsoluteUrl
  Type: Boolean
  Specifies whether to show absolute URLs in the output class (true) or not (false). The default value is false.
Return Value

Type: `ConnectApi.ManagedContentVersionCollection`

`getManagedContentByTopicsAndContentKeys(communityId, contentKeys, topics, pageParam, pageSize, language, managedContentType, showAbsoluteUrl)`

Get managed content versions using a list of content keys and content topic names.

API Version

51.0

Available to Guest Users

51.0

Requires Chatter

No

Signature

```java
public static ConnectApi.ManagedContentVersionCollection getManagedContentByTopicsAndContentKeys(String communityId, List<String> contentKeys, List<String> topics, Integer pageParam, Integer pageSize, String language, String managedContentType, Boolean showAbsoluteUrl)
```

Parameters

`communityId`
- Type: `String`
- ID of the Experience Cloud site.

`contentKeys`
- Type: `List<String>`
- List of up to 50 content keys for the managed content. A content key is a universally unique identifier (UUID) such as `MCA4CCV5QS2BAB5H7YRCRPTCWQZQ`.

`topics`
- Type: `List<String>`
- Comma-separated list of up to 15 content topic names.

`pageParam`
- Type: `Integer`
- Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

`pageSize`
- Type: `Integer`
- Specifies the number of items per page. Valid values are from 1 through 250. For performance reasons, we recommend 25 or fewer items per page. If you pass in `null`, the default size is 25.
language
  Type: String
  Language locale for the managed content, for example, en_US. If the requested translation isn’t available, the language defaults to the context user’s language. If the context user’s language isn’t available, the language defaults to the content type’s original language.

managedContentType
  Type: String
  Developer name of the content type, such as cms_document or cms_image.

showAbsoluteUrl
  Type: Boolean
  Specifies whether to show absolute URLs in the output class (true) or not (false). The default value is false.

Return Value
Type: ConnectApi.ManagedContentVersionCollection

getManagedContentByTopicsAndIds(communityId, managedContentIds, topics, pageParam, pageSize, language, managedContentType)
Get managed content versions using a list of managed content IDs and content topic names.

API Version
47.0

Available to Guest Users
47.0

Requires Chatter
No

Signature
public static ConnectApi.ManagedContentVersionCollection getManagedContentByTopicsAndIds(String communityId, List<String> managedContentIds, List<String> topics, Integer pageParam, Integer pageSize, String language, String managedContentType)

Parameters

communityId
  Type: String
  ID of the Experience Cloud site.

managedContentIds
  Type: List<String>
Comma-separated list of managed content IDs.

topics
  Type: List<String>
  Comma-separated list of up to 15 content topic names.

pageParam
  Type: Integer
  Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
  Type: Integer
  Specifies the number of items per page. Valid values are from 1 through 250. For performance reasons, we recommend 25 or fewer items per page. If you pass in null, the default size is 25.

language
  Type: String
  Language locale for the managed content, for example, en_US. If the requested translation isn’t available, the language defaults to the context user’s language. If the context user’s language isn’t available, the language defaults to the content type’s original language.

managedContentType
  Type: String
  Developer name of the content type, such as cms_document or cms_image.

Return Value
Type: ConnectApi.ManagedContentVersionCollection

getManagedContentByTopicsAndIds(communityId, managedContentIds, topics, pageParam, pageSize, language, managedContentType, showAbsoluteUrl)
Get managed content versions with absolute URLs using a list of managed content IDs and content topic names.

API Version
50.0

Available to Guest Users
50.0

Requires Chatter
No

Signature
public static ConnectApi.ManagedContentVersionCollection getManagedContentByTopicsAndIds(String communityId, List<String> managedContentIds,
List<String> topics, Integer pageParam, Integer pageSize, String language, String managedContentType, Boolean showAbsoluteUrl)

Parameters

communityId
   Type: String
   ID of the Experience Cloud site.

managedContentIds
   Type: List<String>
   Comma-separated list of managed content IDs.

topics
   Type: List<String>
   Comma-separated list of up to 15 content topic names.

pageParam
   Type: Integer
   Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
   Type: Integer
   Specifies the number of items per page. Valid values are from 1 through 250. For performance reasons, we recommend 25 or fewer items per page. If you pass in null, the default size is 25.

language
   Type: String
   Language locale for the managed content, for example, en_US. If the requested translation isn’t available, the language defaults to the context user’s language. If the context user’s language isn’t available, the language defaults to the content type’s original language.

managedContentType
   Type: String
   Developer name of the content type, such as cms_document or cms_image.

showAbsoluteUrl
   Type: Boolean
   Specifies whether to show absolute URLs in the output class (true) or not (false). The default value is false.

Return Value

Type: ConnectApi.ManagedContentVersionCollection

getManagedContentSpace(contentSpaceId)

Get a managed content space.

API Version

55.0
Requires Chatter
No

Signature
public static ConnectApi.ManagedContentSpace getManagedContentSpace(String contentSpaceId)

Parameters
contentSpaceId
Type: String
ID of the managed content space.

Return Value
Type: ConnectApi.ManagedContentSpace

ManagedContentDelivery Class
Get collection items. Get a managed content channel. Get managed content.

Namespace
ConnectApi

ManagedContentDelivery Methods
The following are methods for ManagedContentDelivery. All methods are static.

IN THIS SECTION:
  - getCollectionItemsForChannel(channelId, collectionKeyOrId, language)
    Get collection items for a channel.
  - getCollectionItemsForSite(siteId, collectionKeyOrId, language)
    Get collection items for an Experience Cloud site.
  - getManagedContentChannel(channelId)
    Get a managed content channel.
  - getManagedContentForChannel(channelId, contentKeyOrId, showAbsoluteUrl)
    Get a piece of published content for a channel.
  - getManagedContentForChannel(channelId, contentKeyOrId, language, showAbsoluteUrl)
    Get a piece of published content in a specified language for a channel.
  - getManagedContentForChannel(channelId, contentKeyOrId, language, showAbsoluteUrl, referenceDepth, expandReferences, referencesAsList)
    Get a piece of published content in a specified language with references for a channel.
get Managed Content for Site(siteId, contentKeyOrId, showAbsoluteUrl)
Get a piece of published content for an Experience Cloud site.

get Managed Content for Site(siteId, contentKeyOrId, language, showAbsoluteUrl)
Get a piece of published content in a specified language for an Experience Cloud site.

get Managed Content for Site(siteId, contentKeyOrId, language, showAbsoluteUrl, referenceDepth, expandReferences, referencesAsList)
Get a piece of published content in a specified language with references for an Experience Cloud site.

get Managed Contents for Channel(channelId, managedContentIds, contentKeys, contentTypeFQN, language, publishStartDate, publishEndDate, includeContentBody, referenceDepth, expandReferences, referencesAsList, pageParam, pageSize, showAbsoluteUrl)
Get a collection of published contents for a channel.

get Managed Contents for Site(siteId, managedContentIds, contentKeys, contentTypeFQN, language, publishStartDate, publishEndDate, includeContentBody, referenceDepth, expandReferences, referencesAsList, pageParam, pageSize, showAbsoluteUrl)
Get a collection of published contents for an Experience Cloud site.

get Collection Items for Channel(channelId, collectionKeyOrId, language)
Get collection items for a channel.

API Version
56.0

Available to Guest Users
56.0

Requires Chatter
No

Signature
public static ConnectApi.ManagedContentCollectionItems
getCollectionItemsForChannel(String channelId, String collectionKeyOrId, String language)

Parameters
channelId
Type: String
ID of the channel.

collectionKeyOrId
Type: String
Collection key or ID of the collection. A collection key is a unique identifier such as MCA4CV5QS2BABS7YRCRPTCIGZQ.

language
Type: String
Language locale for the managed content, for example, en-US. If the requested translation isn’t available, the language defaults to the channel or site’s default language. If the channel or site’s default language isn’t available, the language defaults to the primary language of the content space.

Return Value
Type: ConnectApi.ManagedContentCollectionItems

getCollectionItemsForSite(siteId, collectionKeyOrId, language)
Get collection items for an Experience Cloud site.

API Version
56.0

Available to Guest Users
56.0

Requires Chatter
No

Signature
public static ConnectApi.ManagedContentCollectionItems getCollectionItemsForSite(String siteId, String collectionKeyOrId, String language)

Parameters
siteId
Type: String
ID for the Experience Cloud site.

collectionKeyOrId
Type: String
Collection key or ID of the collection. A collection key is a unique identifier such as MCA4CCV5QS2BAB5H7YCRPTCWGZQ.

language
Type: String
Language locale for the managed content, for example, en-US. If the requested translation isn’t available, the language defaults to the channel or site’s default language. If the channel or site’s default language isn’t available, the language defaults to the primary language of the content space.

Return Value
Type: ConnectApi.ManagedContentCollectionItems
getManagedContentChannel(channelId)
Get a managed content channel.

API Version
54.0

Available to Guest Users
54.0

Requires Chatter
No

Signature
public static ConnectApi.ManagedContentChannelDetail getManagedContentChannel(String channelId)

Parameters
channelId
Type: String
ID of the channel.

Return Value
Type: ConnectApi.ManagedContentChannelDetail

getManagedContentForChannel(channelId, contentKeyOrId, showAbsoluteUrl)
Get a piece of published content for a channel.

API Version
54.0

Available to Guest Users
54.0

Requires Chatter
No
Signature

public static ConnectApi.ManagedContentDeliveryDocument
getManagedContentForChannel(String channelId, String contentKeyOrId, Boolean showAbsoluteUrl)

Parameters

channelId
  Type: String
  ID of the channel.

contentKeyOrId
  Type: String
  Content key or ID of the content. A content key is a unique identifier such as MCA4CCV5QS2BAB5H7YRCRPTCWGZQ.

showAbsoluteUrl
  Type: Boolean
  For public channels only, specifies whether to return the absolute unauthenticatedUrl in the output class. The default value is false.

Return Value

Type: ConnectApi.ManagedContentDeliveryDocument

Usage

This method returns content only if it's published in the default language of the channel. If you request content that isn’t published in the default language of the channel, you get a ConnectApi.NotFoundException. To get content for a channel in another language use getManagedContentForChannel(channelId, contentKeyOrId, language, showAbsoluteUrl) or getManagedContentForChannel(channelId, contentKeyOrId, language, showAbsoluteUrl, referenceDepth, expandReferences, referencesAsList) on page 1282.

getManagedContentForChannel(channelId, contentKeyOrId, language, showAbsoluteUrl)
Get a piece of published content in a specified language for a channel.

API Version

54.0

Available to Guest Users

54.0

Requires Chatter

No
Signature

```java
public static ConnectApi.ManagedContentDeliveryDocument
getManagedContentForChannel(String channelId, String contentKeyOrId, String language,
Boolean showAbsoluteUrl)
```

Parameters

- `channelId`  
  Type: `String`  
  ID of the channel.
- `contentKeyOrId`  
  Type: `String`  
  Content key or ID of the content. A content key is a unique identifier such as MCA4CCV5QS2BAB5H7YRCRPTCWGZQ.
- `language`  
  Type: `String`  
  Language locale for the managed content, for example, `en_US`. The requested language must be added to the channel, otherwise, you get a `ConnectApi.NotFoundException`. If the requested translation isn’t available, the language defaults to the channel or site’s default language. If the channel or site’s default language isn’t available, the language defaults to the primary language of the content space.
- `showAbsoluteUrl`  
  Type: `Boolean`  
  For public channels only, specifies whether to return the absolute `unauthenticatedUrl` in the output class. The default value is `false`.

Return Value

Type: `ConnectApi.ManagedContentDeliveryDocument`

```java
getManagedContentForChannel(channelId, contentKeyOrId, language,
showAbsoluteUrl, referenceDepth, expandReferences, referencesAsList)
```

Get a piece of published content in a specified language with references for a channel.

API Version

54.0

Available to Guest Users

54.0

Requires Chatter

No
Signature

public static ConnectApi.ManagedContentDeliveryDocument getManagedContentForChannel(String channelId, String contentKeyOrId, String language, Boolean showAbsoluteUrl, Integer referenceDepth, Boolean expandReferences, Boolean referencesAsList)

Parameters

channelId
Type: String
ID of the channel.

contentKeyOrId
Type: String
Content key or ID of the content. A content key is a unique identifier such as MCA4CCV5QS2BAB5H7YRCRPTCWGZQ.

language
Type: String
Language locale for the managed content, for example, en_US. The requested language must be added to the channel, otherwise, you get a ConnectApi.NotFoundException. If the requested translation isn’t available, the language defaults to the channel or site’s default language. If the channel or site’s default language isn’t available, the language defaults to the primary language of the content space.

showAbsoluteUrl
Type: Boolean
For public channels only, specifies whether to return the absolute unauthenticatedUrl in the output class. The default value is false.

referenceDepth
Type: Integer
An integer 0–3 specifying the depth of references. If you specify 0, the references property of the ConnectApi.ManagedContentDeliveryDocument output class is null. If unspecified, the default value is 0.

expandReferences
Type: Boolean
Specifies whether to include details of references (true) or summaries of references (false) in the output class. If unspecified, the default value is false.

referencesAsList
Type: Boolean
Specifies whether to return the references as a list in the referencesList property of the ConnectApi.ManagedContentDeliveryDocument output class (true). If you specify false, the references are returned as key value pairs in the references property. If unspecified, the default value is false.

Return Value

Type: ConnectApi.ManagedContentDeliveryDocument

getManagedContentForSite(siteId, contentKeyOrId, showAbsoluteUrl)
Get a piece of published content for an Experience Cloud site.
API Version
54.0

Available to Guest Users
54.0

Requires Chatter
No

Signature
```
public static ConnectApi.ManagedContentDeliveryDocument getManagedContentForSite(String siteId, String contentKeyOrId, Boolean showAbsoluteUrl)
```

Parameters
- **siteId**
  
  Type: String
  
  ID for the Experience Cloud site.

- **contentKeyOrId**
  
  Type: String
  
  Content key or ID of the content. A content key is a unique identifier such as MCA4CCV5QS2BAB5H7YCRPTCWGZQ.

- **showAbsoluteUrl**
  
  Type: Boolean
  
  For public channels only, specifies whether to return the absolute unauthenticatedUrl in the output class. The default value is `false`.

Return Value

Type: `ConnectApi.ManagedContentDeliveryDocument`

```
getManagedContentForSite(siteId, contentKeyOrId, language, showAbsoluteUrl)
```

Get a piece of published content in a specified language for an Experience Cloud site.
public static ConnectApi.ManagedContentDeliveryDocument getManagedContentForSite(String siteId, String contentKeyOrId, String language, Boolean showAbsoluteUrl)

Parameters

siteId
Type: String
ID for the Experience Cloud site.

contentKeyOrId
Type: String
Content key or ID of the content. A content key is a unique identifier such as MCA4CCV5QS2BAB5H7YRCRPTCWGZQ.

language
Type: String
Language locale for the managed content, for example, en_US. If the requested translation isn’t available, the language defaults to the channel or site’s default language. If the channel or site’s default language isn’t available, the language defaults to the primary language of the content space.

showAbsoluteUrl
Type: Boolean
For public channels only, specifies whether to return the absolute unauthenticatedUrl in the output class. The default value is false.

Return Value
Type: ConnectApi.ManagedContentDeliveryDocument

getManagedContentForSite(siteId, contentKeyOrId, language, showAbsoluteUrl, referenceDepth, expandReferences, referencesAsList)
Get a piece of published content in a specified language with references for an Experience Cloud site.

API Version
54.0

Available to Guest Users
54.0

Requires Chatter
No
Signature

```java
public static ConnectApi.ManagedContentDeliveryDocument getManagedContentForSite(String siteId, String contentKeyOrId, String language, Boolean showAbsoluteUrl, Integer referenceDepth, Boolean expandReferences, Boolean referencesAsList)
```

Parameters

- **siteId**
  - Type: String
  - ID for the Experience Cloud site.

- **contentKeyOrId**
  - Type: String
  - Content key or ID of the content. A content key is a unique identifier such as MCA4CCV5QS2BAB5H7YRCRPCTCWGZQ.

- **language**
  - Type: String
  - Language locale for the managed content, for example, `en_US`. If the requested translation isn’t available, the language defaults to the channel or site’s default language. If the channel or site’s default language isn’t available, the language defaults to the primary language of the content space.

- **showAbsoluteUrl**
  - Type: Boolean
  - For public channels only, specifies whether to return the absolute `unauthenticatedUrl` in the output class. The default value is `false`.

- **referenceDepth**
  - Type: Integer
  - An integer 0–3 specifying the depth of references. If you specify 0, the `references` property of the `ConnectApi.ManagedContentDeliveryDocument` output class is null. If unspecified, the default value is 0.

- **expandReferences**
  - Type: Boolean
  - Specifies whether to include details of references (true) or summaries of references (false) in the output class. If unspecified, the default value is false.

- **referencesAsList**
  - Type: Boolean
  - Specifies whether to return the references as a list in the `referencesList` property of the `ConnectApi.ManagedContentDeliveryDocument` output class (true). If you specify false, the references are returned as key value pairs in the `references` property. If unspecified, the default value is false.

Return Value

- Type: `ConnectApi.ManagedContentDeliveryDocument`

```java
getManagedContentsForChannel(channelId, managedContentIds, contentKeys, contentTypeFQN, language, publishStartDate, publishEndDate,
```
Get a collection of published contents for a channel.

**API Version**
58.0

**Available to Guest Users**
58.0

**Requires Chatter**
No

**Signature**

```java
public static ConnectApi.ManagedContentDeliveryDocumentCollection getManagedContentsForChannel(String channelId, List<String> managedContentIds, List<String> contentKeys, String contentTypeFQN, String language, String publishStartDate, String publishEndDate, Boolean includeContentBody, Integer referenceDepth, Boolean expandReferences, Boolean referencesAsList, Integer pageParam, Integer pageSize, Boolean showAbsoluteUrl)
```

**Parameters**

- **channelId**
  Type: String
  ID of the channel.

- **managedContentIds**
  Type: List<String>
  Comma-separated list of up to 100 managed content IDs. Specify either managed content IDs or content keys.

- **contentKeys**
  Type: List<String>
  Comma-separated list of up to 50 content keys. Specify either managed content IDs or content keys.

- **contentTypeFQN**
  Type: String
  Fully qualified name of the managed content type.

- **language**
  Type: String
  Language locale for the managed content, for example, en_US. If the requested translation isn’t available, the language defaults to the configured fallback language or the channel's default language. If the content isn’t available in the fallback language and the channel's default language, we return an error.

- **publishStartDate**
  Type: String

- **publishEndDate**
  Type: String

- **includeContentBody**
  Type: Boolean

- **referenceDepth**
  Type: Integer

- **expandReferences**
  Type: Boolean

- **referencesAsList**
  Type: Boolean

- **pageParam**
  Type: Integer

- **pageSize**
  Type: Integer

- **showAbsoluteUrl**
  Type: Boolean
ISO 8601 formatted publish start date.

`publishEndDate`
Type: String
ISO 8601 formatted publish end date.

`includeContentBody`
Type: Boolean
Specifies whether to return the content body (true) or the content summary (false). If unspecified, the default value is false.

`referenceDepth`
Type: Integer
An integer 0–3 specifying the depth of references. If you specify 0, the `references` property of the `ConnectApi.ManagedContentDeliveryDocumentCollection` output class is null. If unspecified, the default value is 0.

`expandReferences`
Type: Boolean
Specifies whether to include details of references (true) or summaries of references (false) in the output class. If unspecified, the default value is false.

`referencesAsList`
Type: Boolean
Specifies whether to return the references as a list in the `referencesList` property of the `ConnectApi.ManagedContentDeliveryDocumentCollection` output class (true). If you specify false, the references are returned as key value pairs in the `references` property. If unspecified, the default value is false.

`pageParam`
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

`pageSize`
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 250. If you pass in null, the default size is 25. If you specify true for `expandReferences` or `includeContentBody`, the maximum page size you can specify is 25.

`showAbsoluteUrl`
Type: Boolean
Specifies whether to show absolute URLs in the output class (true) or not (false). The default value is false.

Return Value
Type: `ConnectApi.ManagedContentDeliveryDocumentCollection`

`getManagedContentsForSite(siteId, managedContentIds, contentKeys, contentTypeFQN, language, publishStartDate, publishEndDate, includeContentBody, referenceDepth, expandReferences, referencesAsList, pageParam, pageSize, showAbsoluteUrl)`

Get a collection of published contents for an Experience Cloud site.
API Version

58.0

Available to Guest Users

58.0

Requires Chatter

No

Signature

public static ConnectApi.ManagedContentDeliveryDocumentCollection
getManagedContentsForSite(String siteId, List<String> managedContentIds, List<String> contentKeys, String contentTypeFQN, String language, String publishStartDate, String publishEndDate, Boolean includeContentBody, Integer referenceDepth, Boolean expandReferences, Boolean referencesAsList, Integer pageParam, Integer pageSize, Boolean showAbsoluteUrl)

Parameters

siteId
Type: String

ID for the Experience Cloud site.

managedContentIds
Type: List<String>

Comma-separated list of up to 100 managed content IDs. Specify either managed content IDs or content keys.

contentKeys
Type: List<String>

Comma-separated list of up to 50 content keys. Specify either managed content IDs or content keys.

customTypeFQN
Type: String

Fully qualified name of the managed content type.

language
Type: String

Language locale for the managed content, for example, en_US. If the requested translation isn’t available, the language defaults to the configured fallback language or the channel’s default language. If the content isn’t available in the fallback language and the channel’s default language, we return an error.

publishStartDate
Type: String

ISO 8601 formatted publish start date.

publishEndDate
Type: String

ISO 8601 formatted publish end date.
**includeContentBody**
Type: Boolean
Specifies whether to return the content body (true) or the content summary (false). If unspecified, the default value is false.

**referenceDepth**
Type: Integer
An integer 0–3 specifying the depth of references. If you specify 0, the references property of the ConnectApi.ManagedContentDeliveryDocumentCollection output class is null. If unspecified, the default value is 0.

**expandReferences**
Type: Boolean
Specifies whether to include details of references (true) or summaries of references (false) in the output class. If unspecified, the default value is false.

**referencesAsList**
Type: Boolean
Specifies whether to return the references as a list in the referencesList property of the ConnectApi.ManagedContentDeliveryDocumentCollection output class (true). If you specify false, the references are returned as key value pairs in the references property. If unspecified, the default value is false.

**pageParam**
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

**pageSize**
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 250. If you pass in null, the default size is 25. If you specify true for expandReferences or includeContentBody, the maximum page size you can specify is 25.

**showAbsoluteUrl**
Type: Boolean
Specifies whether to show absolute URLs in the output class (true) or not (false). The default value is false.

**Return Value**
Type: ConnectApi.ManagedContentDeliveryDocumentCollection

**Retired ManagedContentDelivery Methods**
These methods for ManagedContentDelivery are retired.

_IN THIS SECTION:_

`getManagedContentsForChannel(channelId, managedContentIds, contentKeys, contentTypeFQN, language, publishStartDate, publishEndDate, includeContentBody, referenceDepth, expandReferences, referencesAsList, pageParam, pageSize)`
Get a collection of published contents for a channel.

`getManagedContentsForSite(siteId, managedContentIds, contentKeys, contentTypeFQN, language, publishStartDate, publishEndDate, includeContentBody, referenceDepth, expandReferences, referencesAsList, pageParam, pageSize)`
Get a collection of published contents for an Experience Cloud site.
getManagedContentsForChannel(channelId, managedContentIds, contentKeys, contentTypeFQN, language, publishStartDate, publishEndDate, includeContentBody, referenceDepth, expandReferences, referencesAsList, pageParam, pageSize)

Get a collection of published contents for a channel.

API Version
55.0—57.0
In version 58.0 and later, use getManagedContentsForChannel(channelId, managedContentIds, contentKeys, contentTypeFQN, language, publishStartDate, publishEndDate, includeContentBody, referenceDepth, expandReferences, referencesAsList, pageParam, pageSize, showAbsoluteUrl).

Available to Guest Users
55.0—57.0

Requires Chatter
No

Signature
public static ConnectApi.ManagedContentDeliveryDocumentCollection
getManagedContentsForChannel(String channelId, List<String> managedContentIds, List<String> contentKeys, String contentTypeFQN, String language, String publishStartDate, String publishEndDate, Boolean includeContentBody, Integer referenceDepth, Boolean expandReferences, Boolean referencesAsList, Integer pageParam, Integer pageSize)

Parameters

channelId
Type: String
ID of the channel.

managedContentIds
Type: List<String>
Comma-separated list of up to 100 managed content IDs. Specify either managed content IDs or content keys.

ccontentKeys
Type: List<String>
Comma-separated list of up to 50 content keys. Specify either managed content IDs or content keys.

contentTypeFQN
Type: String
Fully qualified name of the managed content type.
language
  Type: String
  Language locale for the managed content, for example, en_US. If the requested translation isn’t available, the language defaults to the configured fallback language or the channel’s default language. If the content isn’t available in the fallback language and the channel’s default language, we return an error.

publishStartDate
  Type: String
  ISO 8601 formatted publish start date.

publishEndDate
  Type: String
  ISO 8601 formatted publish end date.

includeContentBody
  Type: Boolean
  Specifies whether to return the content body (true) or the content summary (false). If unspecified, the default value is false.

referenceDepth
  Type: Integer
  An integer 0–3 specifying the depth of references. If you specify 0, the references property of the ConnectApi.ManagedContentDeliveryDocumentCollection output class is null. If unspecified, the default value is 0.

expandReferences
  Type: Boolean
  Specifies whether to include details of references (true) or summaries of references (false) in the output class. If unspecified, the default value is false.

referencesAsList
  Type: Boolean
  Specifies whether to return the references as a list in the referencesList property of the ConnectApi.ManagedContentDeliveryDocumentCollection output class (true). If you specify false, the references are returned as key value pairs in the references property. If unspecified, the default value is false.

pageParam
  Type: Integer
  Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
  Type: Integer
  Specifies the number of items per page. Valid values are from 1 through 250. If you pass in null, the default size is 25. If you specify true for expandReferences or includeContentBody, the maximum page size you can specify is 25.

Return Value
Type: ConnectApi.ManagedContentDeliveryDocumentCollection

getManagedContentsForSite(siteId, managedContentIds, contentKeys, contentTypeFQN, language, publishStartDate, publishEndDate,
**getManagedContentsForSite**

Get a collection of published contents for an Experience Cloud site.

**API Version**

55.0—57.0

In version 58.0 and later, use `getManagedContentsForSite(String siteId, List<String> managedContentIds, List<String> contentKeys, String contentTypeFQN, String language, String publishStartDate, String publishEndDate, Boolean includeContentBody, Integer referenceDepth, expandReferences, Boolean referencesAsList, Integer pageParam, Integer pageSize, showAbsoluteUrl).

**Available to Guest Users**

55.0—57.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.ManagedContentDeliveryDocumentCollection getManagedContentsForSite(String siteId, List<String> managedContentIds, List<String> contentKeys, String contentTypeFQN, String language, String publishStartDate, String publishEndDate, Boolean includeContentBody, Integer referenceDepth, Boolean expandReferences, Boolean referencesAsList, Integer pageParam, Integer pageSize)
```

**Parameters**

- `siteId`
  Type: `String`
  ID for the Experience Cloud site.

- `managedContentIds`
  Type: `List<String>`
  Comma-separated list of up to 100 managed content IDs. Specify either managed content IDs or content keys.

- `contentKeys`
  Type: `List<String>`
  Comma-separated list of up to 50 content keys. Specify either managed content IDs or content keys.

- `contentTypeFQN`
  Type: `String`
  Fully qualified name of the managed content type.

- `language`
  Type: `String`
  Language locale for the managed content, for example, `en_US`. If the requested translation isn't available, the language defaults to the configured fallback language or the channel's default language. If the content isn't available in the fallback language and the channel's default language, we return an error.
publishStartDate
  Type: String
  ISO 8601 formatted publish start date.

publishEndDate
  Type: String
  ISO 8601 formatted publish end date.

includeContentBody
  Type: Boolean
  Specifies whether to return the content body (true) or the content summary (false). If unspecified, the default value is false.

referenceDepth
  Type: Integer
  An integer 0–3 specifying the depth of references. If you specify 0, the references property of the
  ConnectApi.ManagedContentDeliveryDocumentCollection output class is null. If unspecified, the default value is 0.

expandReferences
  Type: Boolean
  Specifies whether to include details of references (true) or summaries of references (false) in the output class. If unspecified, the default value is false.

referencesAsList
  Type: Boolean
  Specifies whether to return the references as a list in the referencesList property of the
  ConnectApi.ManagedContentDeliveryDocumentCollection output class (true). If you specify false, the
  references are returned as key value pairs in the references property. If unspecified, the default value is false.

pageParam
  Type: Integer
  Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
  Type: Integer
  Specifies the number of items per page. Valid values are from 1 through 250. If you pass in null, the default size is 25. If you specify true for expandReferences or includeContentBody, the maximum page size you can specify is 25.

Return Value
Type: ConnectApi.ManagedContentDeliveryDocumentCollection

ManagedTopics Class
Get managed topics in an Experience Cloud site. Create, delete, and reorder managed topics.

Namespace
ConnectApi
ManagedTopics Methods

The following are methods for ManagedTopics. All methods are static.

IN THIS SECTION:

createManagedTopic(communityId, recordId, managedTopicType)
Create a managed topic of a specific type for an Experience Cloud site.

createManagedTopic(communityId, recordId, managedTopicType, parentId)
Create a child managed topic for an Experience Cloud site.

createManagedTopicByName(communityId, name, managedTopicType)
Create a managed topic of a specific type by name for an Experience Cloud site.

createManagedTopicByName(communityId, name, managedTopicType, parentId)
Create a child managed topic by name for an Experience Cloud site.

deleteManagedTopic(communityId, managedTopicId)
Delete a managed topic from an Experience Cloud site.

getManagedTopic(communityId, managedTopicId)
Get a managed topic in an Experience Cloud site.

getManagedTopic(communityId, managedTopicId, depth)
Get a managed topic, including its parent and children managed topics, in an Experience Cloud site.

getManagedTopics(communityId)
Get the featured and navigational managed topics for an Experience Cloud site.

getManagedTopics(communityId, managedTopicType)
Get managed topics of the specified type for an Experience Cloud site.

getManagedTopics(communityId, managedTopicType, depth)
Get managed topics of the specified type, including their parent and children managed topics, in an Experience Cloud site.

getManagedTopics(communityId, managedTopicType, recordIds, depth)
Get managed topics of the specified type, including their parent and children managed topics, that are associated with topics in an Experience Cloud site.

getManagedTopics(communityId, managedTopicType, pageParam, pageSize)
Get a page of managed topics.

reorderManagedTopics(communityId, managedTopicPositionCollection)
Reorder the relative positions of managed topics in an Experience Cloud site.

createManagedTopic(communityId, recordId, managedTopicType)
Create a managed topic of a specific type for an Experience Cloud site.

API Version

32.0

Requires Chatter

No
public static ConnectApi.ManagedTopic createManagedTopic(String communityId, String recordId, ConnectApi.ManagedTopicType managedTopicType)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

recordId
Type: String
ID of the topic.

managedTopicType
Type: ConnectApi.ManagedTopicType
Specify the type of managed topic.

- Content—Topics that are associated with native content.
- Featured—Topics that are featured, for example, on the Experience Cloud site home page, but don't provide overall navigation.
- Navigational—Topics that display in a navigational menu in the Experience Cloud site.

A topic can be associated with all three managed topic types, so a topic can be a Featured, Navigational, and Content topic.

You can create up to 25 Featured and 5,000 Content topics. You can create up to eight levels of Navigational managed topics with 25 top-level topics and 10 children topics per level for a maximum of 2,775 Navigational topics.

Return Value
Type: ConnectApi.ManagedTopic

Usage
Only community managers (users with the Create and Set Up Experiences or Manage Experiences permission) can create managed topics.

createManagedTopic(communityId, recordId, managedTopicType, parentId)
Create a child managed topic for an Experience Cloud site.

API Version
35.0

Requires Chatter
No
Signature

public static ConnectApi.ManagedTopic createManagedTopic(String communityId, String recordId, ConnectApi.ManagedTopicType managedTopicType, String parentId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

recordId
Type: String
ID of the topic.

managedTopicType
Type: ConnectApi.ManagedTopicType
Specify Navigational for the type of managed topic to create a child managed topic.
You can create up to 25 Featured and 5,000 Content topics. You can create up to eight levels of Navigational managed topics with 25 top-level topics and 10 children topics per level for a maximum of 2,775 Navigational topics.

parentId
Type: String
ID of the parent managed topic.
You can create up to eight levels (parent, direct children, their children, etc.) of managed topics and up to 10 children managed topics per managed topic.

Return Value

Type: ConnectApi.ManagedTopic

Usage

Only community managers (users with the Create and Set Up Experiences or Manage Experiences permission) can create managed topics.

createManagedTopicByName(communityId, name, managedTopicType)
Create a managed topic of a specific type by name for an Experience Cloud site.

API Version

32.0

Requires Chatter

No
Signature

```java
public static ConnectApi.ManagedTopic createManagedTopicByName(String communityId,
String name, ConnectApi.ManagedTopicType managedTopicType)
```

Parameters

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **name**
  - Type: `String`
  - Name of the topic.

- **managedTopicType**
  - Type: `ConnectApi.ManagedTopicType`
  - Specify the type of managed topic.
    - **Content**—Topics that are associated with native content.
    - **Featured**—Topics that are featured, for example, on the Experience Cloud site home page, but don’t provide overall navigation.
    - **Navigational**—Topics that display in a navigational menu in the Experience Cloud site.

A topic can be associated with all three managed topic types, so a topic can be a `Featured`, `Navigational`, and `Content` topic.

You can create up to 25 `Featured` and 5,000 `Content` topics. You can create up to eight levels of `Navigational` managed topics with 25 top-level topics and 10 children topics per level for a maximum of 2,775 `Navigational` topics.

Return Value

- **Type**: `ConnectApi.ManagedTopic`

Usage

Only community managers (users with the Create and Set Up Experiences or Manage Experiences permission) can create managed topics.

```java
createManagedTopicByName(communityId, name, managedTopicType, parentId)
```

Create a child managed topic by name for an Experience Cloud site.

API Version

- **35.0**

Requires Chatter

- **No**
Signature

```java
public static ConnectApi.ManagedTopic createManagedTopicByName(String communityId, String name, ConnectApi.ManagedTopicType managedTopicType, String parentId)
```

Parameters

`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

`name`
Type: `String`
Name of the topic.

managedTopicType
Type: `ConnectApi.ManagedTopicType`
Specify `Navigational` for the type of managed topic to create a child managed topic.

You can create up to 25 `Featured` and 5,000 `Content` topics. You can create up to eight levels of `Navigational` managed topics with 25 top-level topics and 10 children topics per level for a maximum of 2,775 `Navigational` topics.

`parentId`
Type: `String`
ID of the parent managed topic.

You can create up to eight levels (parent, direct children, their children, etc.) of managed topics and up to 10 children managed topics per managed topic.

Return Value
Type: `ConnectApi.ManagedTopic`

Usage
Only community managers (users with the Create and Set Up Experiences or Manage Experiences permission) can create managed topics.

`deleteManagedTopic(communityId, managedTopicId)`
Delete a managed topic from an Experience Cloud site.

API Version
32.0

Requires Chatter
No

Signature

```java
public static deleteManagedTopic(String communityId, String managedTopicId)
```
Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

managedTopicId
Type: String
ID of managed topic.

Return Value
Type: Void

Usage
Only community managers (users with the Create and Set Up Experiences or Manage Experiences permission) can delete managed topics.

getManagedTopic(communityId, managedTopicId)
Get a managed topic in an Experience Cloud site.

API Version
32.0

Available to Guest Users
32.0

Requires Chatter
No

Signature
public static ConnectApi.ManagedTopic getManagedTopic(String communityId, String managedTopicId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

managedTopicId
Type: String
ID of managed topic.
Return Value
Type: `ConnectApi.ManagedTopic`

`getManagedTopic(communityId, managedTopicId, depth)`
Get a managed topic, including its parent and children managed topics, in an Experience Cloud site.

API Version
35.0

Available to Guest Users
35.0

Requires Chatter
No

Signature
`public static ConnectApi.ManagedTopic getManagedTopic(String communityId, String managedTopicId, Integer depth)`

Parameters
- `communityId`  
  Type: `String`  
  ID for an Experience Cloud site, internal, or `null`.
- `managedTopicId`  
  Type: `String`  
  ID of managed topic.
- `depth`  
  Type: `Integer`  
  Specify an integer 1–8. If you specify 1, the `children` property of the `ConnectApi.ManagedTopic` output class is `null`. If you specify 2, the `children` property of the `ConnectApi.ManagedTopic` output class contains the direct children managed topics, if any, of the managed topic. If you specify 3–8, you get the direct children managed topics and their children managed topics if there are any. If depth isn’t specified, it defaults to 1.

Return Value
Type: `ConnectApi.ManagedTopic`

`getManagedTopics(communityId)`
Get the featured and navigational managed topics for an Experience Cloud site.

To get the content topics for an Experience Cloud site, use `getManagedTopics(communityId, managedTopicType)`. 
API Version
32.0

Available to Guest Users
32.0

Requires Chatter
No

Signature
public static ConnectApi.ManagedTopicCollection getManagedTopics(String communityId)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

Return Value
Type: ConnectApi.ManagedTopicCollection

getManagedTopics(communityId, managedTopicType)
Get managed topics of the specified type for an Experience Cloud site.
ID for an Experience Cloud site, internal, or null.

managedTopicType
Type: ConnectApi.ManagedTopicType
Type of managed topic.
  • Content—Topics that are associated with native content.
  • Featured—Topics that are featured, for example, on the Experience Cloud site home page, but don’t provide overall navigation.
  • Navigational—Topics that display in a navigational menu in the Experience Cloud site.

A topic can be associated with all three managed topic types, so a topic can be a Featured, Navigational, and Content topic.

If you specify Content, up to 50 topics are returned. If you want more than 50 Content topics, use getManagedTopics(communityId, managedTopicType, pageParam, pageSize).

Return Value
Type: ConnectApi.ManagedTopicCollection

getManagedTopics(communityId, managedTopicType, depth)
Get managed topics of the specified type, including their parent and children managed topics, in an Experience Cloud site.

API Version
35.0

Available to Guest Users
35.0

Requires Chatter
No

Signature
public static ConnectApi.ManagedTopicCollection getManagedTopics(String communityId, ConnectApi.ManagedTopicType managedTopicType, Integer depth)

Parameters
communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

managedTopicType
  Type: ConnectApi.ManagedTopicType
  Type of managed topic.
  • Content—Topics that are associated with native content.
• Featured—Topics that are featured, for example, on the Experience Cloud site home page, but don’t provide overall navigation.
• Navigational—Topics that display in a navigational menu in the Experience Cloud site.

A topic can be associated with all three managed topic types, so a topic can be a Featured, Navigational, and Content topic.

depth
Type: Integer
Specify an integer 1–8. If you specify 1, the children property of the ConnectApi.ManagedTopic output class is null. If you specify 2, the children property of the ConnectApi.ManagedTopic output class contains the direct children managed topics, if any, of the managed topic. If you specify 3–8, you get the direct children managed topics and their children managed topics if there are any. If depth isn’t specified, it defaults to 1.

Return Value
Type: ConnectApi.ManagedTopicCollection

getManagedTopics(communityId, managedTopicType, recordIds, depth)
Get managed topics of the specified type, including their parent and children managed topics, that are associated with topics in an Experience Cloud site.

API Version
38.0

Available to Guest Users
38.0

Requires Chatter
No

Signature
public static ConnectApi.ManagedTopicCollection getManagedTopics(String communityId, ConnectApi.ManagedTopicType managedTopicType, List<String> recordIds, Integer depth)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

managedTopicType
Type: ConnectApi.ManagedTopicType
Type of managed topic.
• Content—Topics that are associated with native content.
• Featured—Topics that are featured, for example, on the Experience Cloud site home page, but don’t provide overall navigation.
• Navigational—Topics that display in a navigational menu in the Experience Cloud site.

A topic can be associated with all three managed topic types, so a topic can be a Featured, Navigational, and Content topic.

**recordIds**
Type: List<String>
A list of up to 100 topic IDs associated with the managed topics.
If you list more than 10 topic IDs, you can't specify 2–8 for **depth**.

**depth**
Type: Integer
Specify an integer 1–8. If you specify 1, the children property of the ConnectApi.ManagedTopic output class is **null**. If you specify 2, the children property of the ConnectApi.ManagedTopic output class contains the direct children managed topics, if any, of the managed topic. If you specify 3–8, you get the direct children managed topics and their children managed topics if there are any. If depth isn't specified, it defaults to 1.

**Return Value**
Type: ConnectApi.ManagedTopicCollection

**getManagedTopics(communityId, managedTopicType, pageParam, pageSize)**
Get a page of managed topics.

**API Version**
44.0

**Available to Guest Users**
44.0

**Requires Chatter**
No

**Signature**
```
public static ConnectApi.ManagedTopicCollection getManagedTopics(String communityId,
                     ConnectApi.ManagedTopicType managedTopicType, Integer pageParam, Integer pageSize)
```

**Parameters**

**communityId**
Type: String
ID for an Experience Cloud site, internal, or **null**.

**managedTopicType**
Type: ConnectApi.ManagedTopicType
Type of managed topic.
• Content—Topics that are associated with native content.
• Featured—Topics that are featured, for example, on the Experience Cloud site home page, but don’t provide overall navigation.
• Navigational—Topics that display in a navigational menu in the Experience Cloud site.

A topic can be associated with all three managed topic types, so a topic can be a Featured, Navigational, and Content topic.

`pageParam`  
Type: Integer  
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

`pageSize`  
Type: Integer  
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 50.

Return Value  
Type: ConnectApi.ManagedTopicCollection

`reorderManagedTopics(communityId, managedTopicPositionCollection)`  
Reorder the relative positions of managed topics in an Experience Cloud site.

API Version  
32.0

Requires Chatter  
No

Signature  
`public static ConnectApi.ManagedTopicCollection reorderManagedTopics(String communityId, ConnectApi.ManagedTopicPositionCollectionInput managedTopicPositionCollection)`

Parameters  
`communityId`  
Type: String  
ID for an Experience Cloud site, internal, or null.

`managedTopicPositionCollection`  
Type: ConnectApi.ManagedTopicPositionCollectionInput  
A collection of relative positions of managed topics. This collection can include only Featured and Navigational topics and doesn’t have to include all managed topics.

Return Value  
Type: ConnectApi.ManagedTopicCollection
Usage

Only community managers (users with the Create and Set Up Experiences or Manage Experiences permission) can reorder managed topics.

You can reorder parent managed topics or children managed topics with the same parent. If you don't include all managed topics in the `ConnectApi.ManagedTopicPositionCollectionInput`, the managed topics are reordered by respecting the positions indicated in the `ConnectApi.ManagedTopicPositionCollectionInput` and then by pushing down any managed topics that aren't included in the `ConnectApi.ManagedTopicPositionCollectionInput` to the next available position.

Example

If you have these managed topics:

<table>
<thead>
<tr>
<th>Managed Topic</th>
<th>Position</th>
</tr>
</thead>
<tbody>
<tr>
<td>ManagedTopicA</td>
<td>0</td>
</tr>
<tr>
<td>ManagedTopicB</td>
<td>1</td>
</tr>
<tr>
<td>ManagedTopicC</td>
<td>2</td>
</tr>
<tr>
<td>ManagedTopicD</td>
<td>3</td>
</tr>
<tr>
<td>ManagedTopicE</td>
<td>4</td>
</tr>
</tbody>
</table>

And you reorder managed topics by including this information in `ConnectApi.ManagedTopicPositionCollectionInput`:

<table>
<thead>
<tr>
<th>Managed Topic</th>
<th>Position</th>
</tr>
</thead>
<tbody>
<tr>
<td>ManagedTopicD</td>
<td>0</td>
</tr>
<tr>
<td>ManagedTopicE</td>
<td>2</td>
</tr>
</tbody>
</table>

The result is:

<table>
<thead>
<tr>
<th>Managed Topic</th>
<th>Position</th>
</tr>
</thead>
<tbody>
<tr>
<td>ManagedTopicD</td>
<td>0</td>
</tr>
<tr>
<td>ManagedTopicA</td>
<td>1</td>
</tr>
<tr>
<td>ManagedTopicE</td>
<td>2</td>
</tr>
<tr>
<td>ManagedTopicB</td>
<td>3</td>
</tr>
<tr>
<td>ManagedTopicC</td>
<td>4</td>
</tr>
</tbody>
</table>

Retired ManagedTopics Methods

The following methods for `ManagedTopics` are retired.
getManagedTopics(communityId, managedTopicType, recordId, depth)

Get managed topics of the specified type, including their parent and children managed topics, that are associated with a given topic in an Experience Cloud site.

API Version
35.0–37.0

Important: In version 38.0 and later, use getManagedTopics(communityId, managedTopicType, recordIds, depth).

Available to Guest Users
35.0

Requires Chatter
No

Signature

public static ConnectApi.ManagedTopicCollection getManagedTopics(String communityId, ConnectApi.ManagedTopicType managedTopicType, String recordId, Integer depth)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

managedTopicType
Type: ConnectApi.ManagedTopicType
Type of managed topic.
- Content—Topics that are associated with native content.
- Featured—Topics that are featured, for example, on the Experience Cloud site home page, but don’t provide overall navigation.
- Navigational—Topics that display in a navigational menu in the Experience Cloud site.
A topic can be associated with all three managed topic types, so a topic can be a Featured, Navigational, and Content topic.

recordId
Type: String
ID of the topic associated with the managed topics.
depth
Type: Integer
Specify an integer 1–8. If you specify 1, the children property of the ConnectApi.ManagedTopic output class is null. If you specify 2, the children property of the ConnectApi.ManagedTopic output class contains the direct children managed topics, if any, of the managed topic. If you specify 3–8, you get the direct children managed topics and their children managed topics if there are any. If depth isn’t specified, it defaults to 1.

Return Value
Type: ConnectApi.ManagedTopicCollection

MarketingIntegration Class
Get, save, and submit a microsites marketing integration form for an Experience Cloud site.

Namespace
ConnectApi

MarketingIntegration Methods
The following are methods for MarketingIntegration. All methods are static.
MarketingIntegration methods make calls to Marketing Cloud REST APIs to create, query, and insert data to the data extension object. If Marketing Cloud REST APIs return errors, ConnectinApex error messages include the error code and message from Marketing Cloud.

IN THIS SECTION:
getForm(siteId, formId)
Get a marketing integration form for an Experience Cloud site.
saveForm(siteId, formInput)
Save a marketing integration form for an Experience Cloud site.
submitForm(siteId, formId, formSubmissionInput)
Submit a marketing integration form for an Experience Cloud site.

getForm(siteId, formId)
Get a marketing integration form for an Experience Cloud site.

API Version
53.0

Requires Chatter
No
**Signature**

```
public static ConnectApi.Form getForm(String siteId, String formId)
```

**Parameters**

- `siteId`
  - Type: `String`
  - ID for the Experience Cloud site.

- `formId`
  - Type: `String`
  - ID of the form.

**Return Value**

Type: `ConnectApi.Form`

---

**saveForm(siteId, formInput)**

Save a marketing integration form for an Experience Cloud site.

**API Version**

S3.0

**Requires Chatter**

No

**Signature**

```
public static ConnectApi.Form saveForm(String siteId, ConnectApi.FormInput formInput)
```

**Parameters**

- `siteId`
  - Type: `String`
  - ID for the Experience Cloud site.

- `formInput`
  - Type: `ConnectApi.FormInput`
  - A `ConnectApi.FormInput` object to save.

**Return Value**

Type: `ConnectApi.Form`
Usage
This method attempts to create a read-only data extension in Marketing Cloud. A Marketing Cloud admin can change the read-only setting. We recommend keeping the data extension as read-only to maintain schema consistency with the form.

submitForm(siteId, formId, formSubmissionInput)
Submit a marketing integration form for an Experience Cloud site.

API Version
53.0

Available to Guest Users
53.0

Requires Chatter
No

Signature
public static ConnectApi.FormSubmission submitForm(String siteId, String formId, ConnectApi.FormSubmissionInput formSubmissionInput)

Parameters

siteId
 Type: String
 ID for the Experience Cloud site.

formId
 Type: String
 ID of the form.

formSubmissionInput
 Type: ConnectApi.FormSubmissionInput
 A ConnectApi.FormSubmissionInput object to submit.

Return Value
Type: ConnectApi.FormSubmission

Mentions Class
Access information about mentions. A mention is an "@" character followed by a user or group name. When a user or group is mentioned, they receive a notification.
Namespace

ConnectApi

Mentions Methods

The following are methods for Mentions. All methods are static.

IN THIS SECTION:

getMentionCompletions(communityId, q, contextId)
Get the first page of possible users and groups to mention in a feed item body or comment body.

getMentionCompletions(communityId, q, contextId, type, pageParam, pageSize)
Get a page of possible mention proposals of the specified type.

getMentionValidations(communityId, parentId, recordIds, visibility)
Get information about whether the mentions are valid for the context user.

getMentionCompletions(communityId, q, contextId)
Get the first page of possible users and groups to mention in a feed item body or comment body.

API Version

29.0

Requires Chatter

Yes

Signature

public static ConnectApi.MentionCompletionPage getMentionCompletions (String communityId, String q, String contextId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
A search term for matching user and group names. Searching for a group requires a minimum of 2 characters. Searching for a user doesn’t require a minimum number of characters. This parameter does not support wildcards.

customId
Type: String
A feed item ID (for a mention in a comment) or a feed subject ID (for a mention in a feed item) that narrows search results, with more useful results listed first. Use a group ID for groups that allow customers to ensure mention completion results include customers.
Return Value
Type: `ConnectApi.MentionCompletionPage`

Usage
Call this method to generate a page of proposed mentions that a user can choose from when they enter characters after `@` in a feed item body or a comment body.

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestGetMentionCompletions(communityId, q, contextId, result)`
  
  *Apex Developer Guide: Testing ConnectApi Code*

```
getMentionCompletions(communityId, q, contextId, type, pageParam, pageSize)
```
Get a page of possible mention proposals of the specified type.

API Version
29.0

Requires Chatter
Yes

Signature
```
public static ConnectApi.Mentions getMentionCompletions (String communityId, String q, String contextId, ConnectApi.MentionCompletionType type, Integer pageParam, Integer pageSize)
```

Parameters
- `communityId`
  Type: `String`
  ID for an Experience Cloud site, `internal`, or `null`.

- `q`
  Type: `String`
  A search term for matching user and group names. Searching for a group requires a minimum of 2 characters. Searching for a user doesn't require a minimum number of characters. This parameter does not support wildcards.

- `contextId`
  Type: `String`
  A feed item ID (for a mention in a comment) or a feed subject ID (for a mention in a feed item) that narrows search results, with more useful results listed first. Use a group ID for groups that allow customers to ensure mention completion results include customers.
**Type**

Type: `ConnectApi.MentionCompletionType`

Type of mention completion.

- **All**—All mention completions, regardless of the type of record to which the mention refers.
- **Group**—Mention completions for groups.
- **User**—Mention completions for users.

**pageParam**

Type: `Integer`

Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

**pageSize**

Type: `Integer`

Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**Return Value**

Type: `ConnectApi.MentionCompletionPage`

**Usage**

Call this method to generate a page of proposed mentions that a user can choose from when they enter characters after @ in a feed item body or a comment body.

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- `setTestGetMentionCompletions(communityId, q, contextId, type, pageParam, pageSize, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

**getMentionValidations(communityId, parentId, recordIds, visibility)**

Get information about whether the mentions are valid for the context user.

**API Version**

29.0

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.Mentions getMentionValidations(String communityId, String parentId, List<String> recordIds, ConnectApi.FeedItemVisibilityType visibility)
```
Parameters

- **communityId**
  - Type: **String**
  - ID for an Experience Cloud site, internal, or **null**.

- **parentId**
  - Type: **String**
  - The feed item parent ID.

- **recordIds**
  - Type: List<String>
  - A comma-separated list of IDs to be mentioned. The maximum value is 25.

- **visibility**
  - Type: ConnectApi.FeedItemVisibilityType
  - Type of users who can see a feed item.
    - AllUsers—Visibility is not limited to internal users.
    - InternalUsers—Visibility is limited to internal users.

Return Value

- Type: **ConnectApi.MentionValidations**

Usage

Call this method to check whether the record IDs returned from a call to ConnectApi.Mentions.getMentionCompletions are valid for the context user. For example, the context users can’t mention private groups they don’t belong to. If such a group were included in the list of mention validations, the ConnectApi.MentionValidations.hasErrors property would be **true** and the group would have a ConnectApi.MentionValidation.validationStatus of Disallowed.

Mentions Test Methods

These are the test methods for Mentions. All methods are static.

For information about using these methods to test your ConnectApi code, see Testing ConnectApi Code.

**setTestGetMentionCompletions(communityId, q, contextId, result)**

Register a ConnectApi.MentionCompletionPage object to be returned when getMentionCompletions(String, String, String) is called in a test context.

API Version

29.0

Signature

public static Void setTestGetMentionCompletions (String communityId, String q, String contextId, ConnectApi.MentionCompletionPage result)
Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
A search term for matching user and group names. Searching for a group requires a minimum of 2 characters. Searching for a user doesn’t require a minimum number of characters. This parameter does not support wildcards.

contextId
Type: String
A feed item ID (for a mention in a comment) or a feed subject ID (for a mention in a feed item) that narrows search results, with more useful results listed first. Use a group ID for groups that allow customers to ensure mention completion results include customers.

result
Type: ConnectApi.MentionCompletionPage
A ConnectApi.MentionCompletionPage object containing test data.

Return Value
Type: Void

SEE ALSO:
getMentionCompletions(communityId, q, contextId)

setTestGetMentionCompletions(communityId, q, contextId, type, pageParam, pageSize, result)
Register a ConnectApi.MentionCompletionPage object to be returned when getMentionCompletions(String, String, String, ConnectApi.MentionCompletionType, Integer, Integer) is called in a test context.

API Version
29.0

Signature
public static Void setTestGetMentionCompletions(String communityId, String q, String contextId, ConnectApi.MentionCompletionType type, Integer pageParam, Integer pageSize, ConnectApi.MentionCompletionPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.
q
Type: String
A search term for matching user and group names. Searching for a group requires a minimum of 2 characters. Searching for a user doesn’t require a minimum number of characters. This parameter does not support wildcards.

contextId
Type: String
A feed item ID (for a mention in a comment) or a feed subject ID (for a mention in a feed item) that narrows search results, with more useful results listed first. Use a group ID for groups that allow customers to ensure mention completion results include customers.

type
Type: ConnectApi.MentionCompletionType
Type of mention completion.
• All—All mention completions, regardless of the type of record to which the mention refers.
• Group—Mention completions for groups.
• User—Mention completions for users.

pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

result
Type: ConnectApi.MentionCompletionPage
A ConnectApi.MentionCompletionPage object containing test data.

Return Value
Type: Void

SEE ALSO:
getMentionCompletions(communityId, q, contextId, type, pageParam, pageSize)

Missions Class
Export and purge mission activity for users. Get a user’s mission progress. Update mission activity counts for users.

Namespace
ConnectApi

Missions Methods
The following are methods for Missions. All methods are static.
IN THIS SECTION:

exportUserMissionsActivities(communityId, userId)
Export mission activity for a user.

ggetUserMissionsProgress(communityId, userId)
Get mission activity progress for a user.

purgeUserMissionsActivities(communityId, userId)
Start a job to purge mission activity for a user.

purgeUserMissionsActivities(communityId)
Start a job to purge mission activity for all users.

updateUserMissionActivityCount(activityType, activityCount, communityId, userId)
Update the mission activity count for a user.

---

**exportUserMissionsActivities(communityId, userId)**

Export mission activity for a user.

**API Version**

45.0

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.UserMissionActivitiesJob exportUserMissionsActivities(String communityId, String userId)
```

**Parameters**

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- `userId`
  - Type: `String`
  - ID of the user.

**Return Value**

Type: `ConnectApi.UserMissionActivitiesJob`

**Usage**

You can export these activities with this method.

- `FeedItemAnswerAQuestion`—User answered a question.
• FeedItemLikeSomething—User liked a post or comment.
• FeedItemMarkAnswerAsBest—User marked an answer as the best answer.
• FeedItemPostQuestion—User posted a question.
• FeedItemReceiveAComment—User received a comment on a post.
• FeedItemReceiveALike—User received a like on a post or comment.
• FeedItemReceiveAnAnswer—User received an answer to a question.
• FeedItemWriteAComment—User commented on a post.
• FeedItemWriteAPost—User made a post.
• FeedItemYourAnswerMarkedBest—User’s answer was marked as the best answer.

getUserMissionsProgress(communityId, userId)
Get mission activity progress for a user.

API Version
46.0

Requires Chatter
Yes

Signature
public static ConnectApi.UserMissionActivityCollection getUserMissionsProgress(String communityId, String userId)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

userId
Type: String
ID of the user.

Return Value
Type: ConnectApi.UserMissionActivityCollection

purgeUserMissionsActivities(communityId, userId)
Start a job to purge mission activity for a user.

API Version
45.0
Requires Chatter
Yes

Signature
public static ConnectApi.UserMissionActivitiesJob purgeUserMissionsActivities(String communityId, String userId)

Parameters
communityId
    Type: String
    ID for an Experience Cloud site, internal, or null.

userId
    Type: String
    ID of the user.

Return Value
Type: ConnectApi.UserMissionActivitiesJob

Usage
This method purges these activities.
- FeedItemAnswerAQuestion—User answered a question.
- FeedItemLikeSomething—User liked a post or comment.
- FeedItemMarkAnswerAsBest—User marked an answer as the best answer.
- FeedItemPostQuestion—User posted a question.
- FeedItemReceiveAComment—User received a comment on a post.
- FeedItemReceiveALike—User received a like on a post or comment.
- FeedItemReceiveAnAnswer—User received an answer to a question.
- FeedItemWriteAComment—User commented on a post.
- FeedItemWriteAPost—User made a post.
- FeedItemYourAnswerMarkedBest—User’s answer was marked as the best answer.

purgeUserMissionsActivities(communityId)
Start a job to purge mission activity for all users.

API Version
49.0

Requires Chatter
Yes
public static ConnectApi.UserMissionActivitiesJob purgeUserMissionsActivities(String communityId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

Return Value
Type: ConnectApi.UserMissionActivitiesJob

Usage
This method purges these activities.

- FeedItemAnswerAQuestion — User answered a question.
- FeedItemLikeSomething — User liked a post or comment.
- FeedItemMarkAnswerAsBest — User marked an answer as the best answer.
- FeedItemPostQuestion — User posted a question.
- FeedItemReceiveAComment — User received a comment on a post.
- FeedItemReceiveALike — User received a like on a post or comment.
- FeedItemReceiveAnAnswer — User received an answer to a question.
- FeedItemWriteAComment — User commented on a post.
- FeedItemWriteAPost — User made a post.
- FeedItemYourAnswerMarkedBest — User’s answer was marked as the best answer.

updateUserMissionActivityCount(activityType, activityCount, communityId, userId)
Update the mission activity count for a user.

API Version
45.0

Requires Chatter
Yes

Signature
public static ConnectApi.UserMissionActivityStatus updateUserMissionActivityCount(ConnectApi.UserMissionActivityType activityType, Integer activityCount, String communityId, String userId)
Parameters

**activityType**
   Type: **ConnectApi.UserMissionActivityType**
   Type of mission activity for a user. Values are:
   - **FeedItemAnswerAQuestion**—User answered a question.
   - **FeedItemLikeSomething**—User liked a post or comment.
   - **FeedItemMarkAnswerAsBest**—User marked an answer as the best answer.
   - **FeedItemPostQuestion**—User posted a question.
   - **FeedItemReceiveAComment**—User received a comment on a post.
   - **FeedItemReceiveALike**—User received a like on a post or comment.
   - **FeedItemReceiveAnAnswer**—User received an answer to a question.
   - **FeedItemWriteAComment**—User commented on a post.
   - **FeedItemWriteAPost**—User made a post.
   - **FeedItemYourAnswerMarkedBest**—User’s answer was marked as the best answer.

**activityCount**
   Type: **Integer**
   Number of mission activities of the specified type for the user.

**communityId**
   Type: **String**
   ID for an Experience Cloud site, internal, or null.

**userId**
   Type: **String**
   ID of the user.

Return Value
Type: **ConnectApi.UserMissionActivityStatus**

**NamedCredentials Class**

Create, refresh, get, delete, replace, and update credentials. Create and get external credentials. Create and get named credentials. Get the URL for the OAuth token flow for an external credential.

**Note:** Managed packages can access only the named credentials and external credentials that are included in or created from the package’s Apex code. If a managed package tries to access non-packaged named credentials and external credentials that a Salesforce admin created in the org, an error occurs.

**Namespace**

ConnectApi

**NamedCredentials Methods**

The following are methods for **NamedCredentials**. All methods are static.
IN THIS SECTION:

createCredential(requestBody)
Create a credential.

createCredential(requestBody, action)
Refresh an OAuth or AWS Roles Anywhere credential.

createExternalCredential(requestBody)
Create an external credential.

createNamedCredential(requestBody)
Create a named credential.

deleteCredential(externalCredential, principalName, principalType)
Delete a credential.

deleteCredential(externalCredential, principalName, principalType, authenticationParameters)
Delete a credential with authentication parameters.

getCredential(externalCredential, principalName, principalType)
Get a credential.

getExternalCredential(developerName)
Get an external credential, including the named credentials and principals associated with it and the type and status of each principal.

getExternalCredentials()
Get external credentials that the user can authenticate to.

getNamedCredential(developerName)
Get a named credential.

getNamedCredentials()
Get a list of named credentials for the org.

getOAuthCredentialAuthUrl(requestBody)
Get the URL for the OAuth token flow for an external credential.

patchCredential(requestBody)
Update custom credentials.

updateCredential(requestBody)
Replace a credential.

createCredential (requestBody)
Create a credential.

API Version
56.0

Requires Chatter
No
public static ConnectApi.Credential createCredential(ConnectApi.CredentialInput requestBody)

Parameters
requestBody
Type: ConnectApi.CredentialInput
A ConnectApi.CredentialInput class.

Return Value
Type: ConnectApi.Credential

createCredential(requestBody, action)
Refresh an OAuth or AWS Roles Anywhere credential.

API Version
58.0

Requires Chatter
No

Signature
public static ConnectApi.Credential createCredential(ConnectApi.CredentialInput requestBody, ConnectApi.CreateCredentialAction action)

Parameters
requestBody
Type: ConnectApi.CredentialInput
A ConnectApi.CredentialInput class.
action
Type: ConnectApi.CreateCredentialAction
Action to take when creating the credential. Value is:
• Refresh

Return Value
Type: ConnectApi.Credential

createExternalCredential(requestBody)
Create an external credential.
API Version
58.0

Requires Chatter
No

Signature
public static ConnectApi.ExternalCredential createExternalCredential(ConnectApi.ExternalCredentialInput requestBody)

Parameters
requestBody
  Type: ConnectApi.ExternalCredentialInput
  Input used to create or update an external credential.

Return Value
Type: ConnectApi.ExternalCredential

createNamedCredential(requestBody)
Create a named credential.

API Version
58.0

Requires Chatter
No

Signature
public static ConnectApi.NamedCredential createNamedCredential(ConnectApi.NamedCredentialInput requestBody)

Parameters
requestBody
  Type: ConnectApi.NamedCredentialInput
  Input used to create or update a named credential.

Return Value
Type: ConnectApi.NamedCredential
deleteCredential(externalCredential, principalName, principalType)
Delete a credential.

API Version
56.0

Requires Chatter
No

Signature
public static Void deleteCredential(String externalCredential, String principalName, ConnectApi.CredentialPrincipalType principalType)

Parameters

externalCredential
Type: String
Fully qualified developer name of the external credential.

principalName
Type: String
Name of the external credential named principal.

principalType
Type: ConnectApi.CredentialPrincipalType
Type of credential principal. Values are:
• AwsStsPrincipal
• NamedPrincipal
• PerUserPrincipal

Return Value
Type: Void

deleteCredential(externalCredential, principalName, principalType, authenticationParameters)
Delete a credential with authentication parameters.

API Version
59.0

Requires Chatter
No
**Signature**

```java
public static Void deleteCredential(String externalCredential, String principalName,
ConnectApi.CredentialPrincipalType principalType, List<String> authenticationParameters)
```

**Parameters**

- `externalCredential`
  - Type: `String`
  - Fully qualified developer name of the external credential.

- `principalName`
  - Type: `String`
  - Name of the external credential named principal.

- `principalType`
  - Type: `ConnectApi.CredentialPrincipalType`
  - Type of credential principal. Values are:
    - `AwsStsPrincipal`
    - `NamedPrincipal`
    - `PerUserPrincipal`

- `authenticationParameters`
  - Type: `List<String>`
  - List of authentication parameters only for custom protocols, for example `myApiKey, myApiSecret`. If unspecified, all credentials are deleted.

**Return Value**

Type: `Void`

**getCredential(externalCredential, principalName, principalType)**

Get a credential.

**API Version**

56.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.Credential getCredential(String externalCredential, String
principalName, ConnectApi.CredentialPrincipalType principalType)
```
Parameters

externalCredential
Type: String
Fully qualified developer name of the external credential.

principalName
Type: String
Name of the external credential named principal.

principalType
Type: ConnectApi.CredentialPrincipalType
Type of credential principal. Values are:
- AwsStsPrincipal
- NamedPrincipal
- PerUserPrincipal

Return Value
Type: ConnectApi.Credential

getExternalCredential('developerName')
Get an external credential, including the named credentials and principals associated with it and the type and status of each principal.

API Version
56.0

Requires Chatter
No

Signature
public static ConnectApi.ExternalCredential getExternalCredential(String 'developerName')

Parameters
developerName
Type: String
Fully qualified developer name of the external credential.

Return Value
Type: ConnectApi.ExternalCredential

getExternalCredentials()
Get external credentials that the user can authenticate to.
API Version
56.0

Requires Chatter
No

Signature
public static ConnectApi.ExternalCredentialList getExternalCredentials()

Return Value
Type: ConnectApi.ExternalCredentialList

getNamedCredential(developerName)
Get a named credential.

API Version
59.0

Requires Chatter
No

Signature
public static ConnectApi.NamedCredential getNamedCredential(String developerName)

Parameters
developerName
Type: String
Fully qualified developer name of the named credential.

Return Value
Type: ConnectApi.NamedCredential

getNamedCredentials()
Get a list of named credentials for the org.

API Version
59.0
Required Chatter
No

Signature

```java
public static ConnectApi.NamedCredentialList getNamedCredentials()
```

Return Value
Type: `ConnectApi.NamedCredentialList`

```java
getOAuthCredentialAuthUrl(requestBody)
```
Get the URL for the OAuth token flow for an external credential.

API Version
56.0

Requires Chatter
No

Signature

```java
public static ConnectApi.OAuthCredentialAuthUrl getOAuthCredentialAuthUrl(ConnectApi.OAuthCredentialAuthUrlInput requestBody)
```

Parameters

- **requestBody**
  Type: `ConnectApi.OAuthCredentialAuthUrlInput`
  A `ConnectApi.OAuthCredentialAuthUrlInput` class indicating the OAuth authentication flow.

Return Value
Type: `ConnectApi.OAuthCredentialAuthUrl`

Usage
Accepts input parameters representing a specific external credential and, optionally, a named principal. Returns the URL a user must visit to begin the authentication flow, ultimately returning authentication tokens to Salesforce. Use this method as part of building a customized or branded user interface to help users initiate authentication.

Example

```java
ConnectApi.OAuthCredentialAuthUrlInput input = new ConnectApi.OAuthCredentialAuthUrlInput();
input.externalCredential = 'MyExternalCredentialDeveloperName';
input.principalType = ConnectApi.CredentialPrincipalType.PerUserPrincipal;
```
input.principalName = 'MyPrincipal'; // Only required when principalType = NamedPrincipal

ConnectApi.OAuthCredentialAuthUrl output =
ConnectApi.NamedCredentials.getOAuthCredentialAuthUrl(input);

String authenticationUrl = output.authenticationUrl; // Redirect users to this URL to authenticate in the browser

**patchCredential** *(requestBody)*

Update custom credentials.
This method updates custom credentials. To replace a credential, use **updateCredential** *(requestBody)* .

**API Version**
$9.0

**Requires Chatter**
No

**Signature**

```
public static ConnectApi.Credential patchCredential(ConnectApi.CredentialInput requestBody)
```

**Parameters**

*requestBody*
Type: **ConnectApi.CredentialInput**
A ConnectApi.CredentialInput class. Only the custom credentials in the input class are updated.

**Return Value**
Type: **ConnectApi.Credential**

**updateCredential** *(requestBody)*

Replace a credential.
This method replaces a credential. To update a credential, use **patchCredential** *(requestBody)* .

**API Version**
$6.0

**Requires Chatter**
No
Signature

```java
public static ConnectApi.Credential updateCredential(ConnectApi.CredentialInput requestBody)
```

Parameters

```java
requestBody
  Type: ConnectApi.CredentialInput
  A ConnectApi.CredentialInput class.
```

Return Value

Type: ConnectApi.Credential

NavigationMenu Class

Get navigation menu items for an Experience Cloud site.

Namespace

ConnectApi

NavigationMenu Methods

These methods are for NavigationMenu. All methods are static.

IN THIS SECTION:

```java
getCommunityNavigationMenu(communityId, navigationLinkSetId, navigationLinkSetDeveloperName, publishStatus, includeImageUrl, addHomeMenuItem, menuItemTypesToSkip)
Get navigation menu items for an Experience Cloud site.

getCommunityNavigationMenu(communityId, navigationLinkSetId, navigationLinkSetDeveloperName, publishStatus, includeImageUrl, addHomeMenuItem, menuItemTypesToSkip, effectiveAccountId)
Get navigation menu items for an Experience Cloud based on an effective account.
```

```java
getCommunityNavigationMenu(communityId, navigationLinkSetId, navigationLinkSetDeveloperName, publishStatus, includeImageUrl, addHomeMenuItem, menuItemTypesToSkip)
Get navigation menu items for an Experience Cloud site.
```

API Version

52.0

Available to Guest Users

52.0
Requires Chatter
No

Signature

public static ConnectApi.NavigationMenuItemCollection getCommunityNavigationMenu(String communityId, String navigationLinkSetId, String navigationLinkSetDeveloperName, ConnectApi.PublishStatus publishStatus, Boolean includeImageUrl, Boolean addHomeMenuItem, List<ConnectApi.NavigationMenuItemType> menuItemTypesToSkip)

Parameters

communityId
Type: String
ID of an Experience Cloud site.

navigationLinkSetId
Type: String
ID of the navigation link set.

navigationLinkSetDeveloperName
Type: String
Developer name of the navigation link set.

publishStatus
Type: ConnectApi.PublishStatus
Publish status of the navigation menu item. Values are:
- Draft
- Live

includeImageUrl
Type: Boolean
Specifies whether to include the image URL with the menu item (true) or not (false).

addHomeMenuItem
Type: Boolean
Specifies whether to add the Home menu item (true) or not (false).

menuItemTypesToSkip
Type: List<ConnectApi.NavigationMenuItemType>
List of menu item types to filter out of the results. Values are:
- DataSourceDriven—Menu items dynamically added from a data source.
- Event—Event, such as logging in, logging out, or switching accounts.
- ExternalLink—URL outside of your site.
- GlobalAction—Lets users create records that aren’t related to other records.
- InternalLink—Relative URL inside your site.
- MenuLabel—Menu label.
- Modal—Modal, such as Account Switcher.
- NavigationalTopic—Dropdown list with links to the navigational topics in your site.
- SalesforceObject—Objects such as accounts, cases, contacts, and custom objects.
- SystemLink—System link, such as a link to Builder, Workspaces, or Setup.

**Return Value**

Type: `ConnectApi.NavigationMenuItemCollection`

**Usage**

Supported navigation menu item types are:

- DataSourceDriven—Menu items dynamically added from a data source.
- Event—Event, such as logging in, logging out, or switching accounts.
- ExternalLink—URL outside of your site.
- GlobalAction—Lets users create records that aren't related to other records.
- InternalLink—Relative URL inside your site.
- MenuLabel—Menu label.
- Modal—Modal, such as Account Switcher.
- NavigationalTopic—Dropdown list with links to the navigational topics in your site.
- SalesforceObject—Objects such as accounts, cases, contacts, and custom objects.
- SystemLink—System link, such as a link to Builder, Workspaces, or Setup.

```java
getCommunityNavigationMenu(String communityId, String navigationLinkSetId, String navigationLinkSetDeveloperName, ConnectApi.PublishStatus publishStatus, Boolean includeImageUrl, Boolean addHomeMenuItem, List<ConnectApi.NavigationMenuItemType> menuItemTypesToSkip, String effectiveAccountId)
```

Get navigation menu items for an Experience Cloud based on an effective account.

**API Version**

54.0

**Available to Guest Users**

54.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.NavigationMenuItemCollection getCommunityNavigationMenu(String communityId, String navigationLinkSetId, String navigationLinkSetDeveloperName, ConnectApi.PublishStatus publishStatus, Boolean includeImageUrl, Boolean addHomeMenuItem, List<ConnectApi.NavigationMenuItemType> menuItemTypesToSkip, String effectiveAccountId)
```
Parameters

communityId
Type: String
ID of an Experience Cloud site.

navigationLinkSetId
Type: String
ID of the navigation link set.

navigationLinkSetDeveloperName
Type: String
Developer name of the navigation link set.

publishStatus
Type: ConnectApi.PublishStatus
Publish status of the navigation menu item. Values are:
- Draft
- Live

includeImageUrl
Type: Boolean
Specifies whether to include the image URL with the menu item (true) or not (false).

addHomeMenuItem
Type: Boolean
Specifies whether to add the Home menu item (true) or not (false).

menuItemTypesToSkip
Type: List<ConnectApi.NavigationMenuItemType>
List of menu item types to filter out of the results. Values are:
- DataSourceDriven—Menu items dynamically added from a data source.
- Event—Event, such as logging in, logging out, or switching accounts.
- ExternalLink—URL outside of your site.
- GlobalAction—Lets users create records that aren’t related to other records.
- InternalLink—Relative URL inside your site.
- MenuLabel—Menu label.
- Modal—Modal, such as Account Switcher.
- NavigationalTopic—Dropdown list with links to the navigational topics in your site.
- SalesforceObject—Objects such as accounts, cases, contacts, and custom objects.
- SystemLink—System link, such as a link to Builder, Workspaces, or Setup.

effectiveAccountId
Type: String
ID of the account for which the request is made. If unspecified, defaults to the account ID for the context user.

Return Value
Type: ConnectApi.NavigationMenuItemCollection
Usage

Supported navigation menu item types are:

- **DataSourceDriven**—Menu items dynamically added from a data source.
- **Event**—Event, such as logging in, logging out, or switching accounts.
- **ExternalLink**—URL outside of your site.
- **GlobalAction**—Lets users create records that aren’t related to other records.
- **InternalLink**—Relative URL inside your site.
- **MenuLabel**—Menu label.
- **Modal**—Modal, such as Account Switcher.
- **NavigationalTopic**—Dropdown list with links to the navigational topics in your site.
- **SalesforceObject**—Objects such as accounts, cases, contacts, and custom objects.
- **SystemLink**—System link, such as a link to Builder, Workspaces, or Setup.

**NextBestAction Class**

Execute recommendation strategies, get recommendations, manage recommendation reactions.

**Namespace**

ConnectApi

**Usage**

Community users can’t access this class. Portal and guest users can access strategies only through the Suggested Actions component.

**NextBestAction Methods**

The following are methods for NextBestAction. All methods are static.

<table>
<thead>
<tr>
<th>Method Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>deleteRecommendationReaction(reactionId)</code></td>
<td>Delete a recommendation reaction.</td>
</tr>
<tr>
<td><code>executeStrategy(strategyName, maxResults, contextRecordId)</code></td>
<td>Execute a strategy.</td>
</tr>
<tr>
<td><code>executeStrategy(strategyName, maxResults, contextRecordId, debugTrace)</code></td>
<td>Execute a strategy and request a trace.</td>
</tr>
<tr>
<td><code>executeStrategy(strategyName, strategyInput)</code></td>
<td>Execute a strategy using an input class.</td>
</tr>
<tr>
<td><code>getRecommendation(recommendationId)</code></td>
<td>Get a recommendation.</td>
</tr>
<tr>
<td><code>getRecommendationReaction(reactionId)</code></td>
<td>Get a recommendation reaction.</td>
</tr>
</tbody>
</table>
getRecommendationReactions(onBehalfOfId, createdById, targetId, contextRecordId, pageParam, pageSize)
Get recommendation reactions.

setRecommendationReaction(reaction)
Record user reactions to recommendations.

deleteRecommendationReaction(reactionId)
Delete a recommendation reaction.

API Version
45.0

Requires Chatter
No

Signature
public static Void deleteRecommendationReaction(String reactionId)

Parameters
reactionId
Type: String
ID of the recommendation reaction.

Return Value
Type: Void

Usage
Users with the Manage Next Best Action Recommendations or Modify All Data permission can delete recommendation reactions.

executeStrategy(strategyName, maxResults, contextRecordId)
Execute a strategy.

API Version
45.0

Available to Guest Users
45.0

Requires Chatter
No
Signature

```java
public static ConnectApi.NBARecommendations executeStrategy(String strategyName, Integer maxResults, String contextRecordId)
```

Parameters

- `strategyName`
  - Type: `String`
  - Name of the strategy.

- `maxResults`
  - Type: `Integer`
  - Maximum number of results. Valid values are from 1 to 25. The default is 3.

- `contextRecordId`
  - Type: `String`
  - ID of the context record. For example, if the next best action is on a case detail page, the ID of the case.

Return Value

Type: `ConnectApi.NBARecommendations`

```java
executeStrategy(strategyName, maxResults, contextRecordId, debugTrace)
```

Execute a strategy and request a trace.

API Version

45.0

Available to Guest Users

45.0

Requires Chatter

No

Signature

```java
public static ConnectApi.NBARecommendations executeStrategy(String strategyName, Integer maxResults, String contextRecordId, Boolean debugTrace)
```

Parameters

- `strategyName`
  - Type: `String`
  - Name of the strategy.

- `maxResults`
  - Type: `Integer`
  - Maximum number of results. Valid values are from 1 to 25. The default is 3.
Maximum number of results. Valid values are from 1 to 25. The default is 3.

**contextRecordId**
Type: String

ID of the context record. For example, if the next best action is on a case detail page, the ID of the case.

**debugTrace**
Type: Boolean

Specifies whether to return trace and debug information in the response (true) or not (false).

### Return Value
Type: `ConnectApi.NBARecommendations`

#### executeStrategy(strategyName, strategyInput)
Execute a strategy using an input class.

### API Version
45.0

### Available to Guest Users
45.0

### Requires Chatter
No

### Signature

```java
public static ConnectApi.NBARecommendations executeStrategy(String strategyName, ConnectApi.NBAStrategyInput strategyInput)
```

### Parameters

- **strategyName**
  Type: String
  Name of the strategy.

- **strategyInput**
  Type: `ConnectApi.NBAStrategyInput`
  A `ConnectApi.NBAStrategyInput` body.

### Return Value
Type: `ConnectApi.NBARecommendations`
**getRecommendation(recommendationId)**
Get a recommendation.

**API Version**
45.0

**Requires Chatter**
No

**Signature**
```
public static ConnectApi.Recommendation getRecommendation(String recommendationId)
```

**Parameters**
- `recommendationId`
  - Type: `String`
  - ID of the recommendation.

**Return Value**
Type: `ConnectApi.Recommendation`

---

**getRecommendationReaction(reactionId)**
Get a recommendation reaction.

**API Version**
45.0

**Requires Chatter**
No

**Signature**
```
public static ConnectApi.RecommendationReaction getRecommendationReaction(String reactionId)
```

**Parameters**
- `reactionId`
  - Type: `String`
  - ID of the recommendation reaction.
Return Value
Type: `ConnectApi.RecommendationReaction`

Usage
Users with the Manage Next Best Action Recommendations or Modify All Data permission can get recommendation reactions.

`getRecommendationReactions(onBehalfOfId, createdById, targetId, contextRecordId, pageParam, pageSize)`
Get recommendation reactions.

API Version
45.0

Requires Chatter
No

Signature
`public static ConnectApi.RecommendationReactions getRecommendationReactions(String onBehalfOfId, String createdById, String targetId, String contextRecordId, Integer pageParam, Integer pageSize)`

Parameters
`onBehalfOfId`
Type: `String`
Use the ID of the user who is indirectly reacting to the recommendation to filter the results.

`createdById`
Type: `String`
Use the ID of the user or record that created the recommendation reaction to filter the results.

`targetId`
Type: `String`
Use the ID of the target to filter the results.

`contextRecordId`
Type: `String`
Use the ID of a context record to filter the results.

`pageParam`
Type: `Integer`
Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

`pageSize`
Type: `Integer`
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.
Return Value
Type: ConnectApi.RecommendationReaction

Usage
Users with the Manage Next Best Action Recommendations or Modify All Data permission can get recommendation reactions.

setRecommendationReaction(reaction)
Record user reactions to recommendations.

API Version
45.0

Available to Guest Users
48.0

Requires Chatter
No

Signature
public static ConnectApi.RecommendationReaction setRecommendationReaction(ConnectApi.RecommendationReactionInput reaction)

Parameters
reaction
Type: ConnectApi.RecommendationReactionInput
A ConnectApi.RecommendationReactionInput object representing a reaction to a recommendation produced by a recommendation strategy.

Return Value
Type: ConnectApi.RecommendationReaction

OmnichannelInventoryService Class
Route orders to inventory locations in Order Management.

Namespace
ConnectApi
OmnichannelInventoryService Methods

The following are methods for OmnichannelInventoryService. All methods are static.

**IN THIS SECTION:**
- `createReservation(createReservationInputRepresentation)`
- `fulfillReservation(fulfillReservationInputRepresentation)`
- `getInventoryAvailability(inventoryAvailabilityInputRepresentation)`
- `getInventoryAvailabilityUploadStatus(uploadId)`
- `getPublishLocationStructureStatus(uploadId)`
- `publishLocationStructure()`
- `releaseReservation(releaseReservationInputRepresentation)`
- `submitInventoryAvailabilityUpload(fileUpload)`
- `transferReservation(transferReservationInputRepresentation)`

---

### `createReservation(createReservationInputRepresentation)`
Create an inventory reservation in Omnichannel Inventory.

**API Version**
51.0

**Requires Chatter**
No

**Signature**

```java
public static ConnectApi.OCICreateReservationOutputRepresentation createReservation(ConnectApi.OCICreateReservationInputRepresentation)
```
Parameters

createReservationInputRepresentation

Type: ConnectApi.OCICreateReservationInputRepresentation

Data to reserve inventory at one or more Omnichannel Inventory locations or location groups.

Return Value

Type: ConnectApi.OCICreateReservationOutputRepresentation

fulfillReservation(fulfillReservationInputRepresentation)

Fulfill one or more inventory reservations.

API Version

$1.0

Requires Chatter

No

Signature

public static ConnectApi.OCIFulfillReservationOutputRepresentation fulfillReservation(ConnectApi.OCIFulfillReservationInputRepresentation fulfillReservationInputRepresentation)

Parameters

fulfillReservationInputRepresentation

Type: ConnectApi.OCIFulfillReservationInputRepresentation

Wraps a list of inventory reservations to fulfill.

Return Value

Type: ConnectApi.OCIFulfillReservationOutputRepresentation

getInventoryAvailability(inventoryAvailabilityInputRepresentation)

Retrieve inventory availability data for one or more products at one or more inventory locations or location groups.

API Version

$1.0

Requires Chatter

No
Signature

```java
public static ConnectApi.OCIGetInventoryAvailabilityOutputRepresentation
getInventoryAvailability(ConnectApi.OCIGetInventoryAvailabilityInputRepresentation
inventoryAvailabilityInputRepresentation)
```

Parameters

`inventoryAvailabilityInputRepresentation`
Type: `ConnectApi.OCIGetInventoryAvailabilityInputRepresentation`
Details of a request to retrieve inventory availability.

Return Value
Type: `ConnectApi.OCIGetInventoryAvailabilityOutputRepresentation`

```java
getInventoryAvailabilityUploadStatus(uploadId)
```

Retrieve the status of an inventory availability upload job.

API Version
5.0

Requires Chatter
No

Signature

```java
public static ConnectApi.OCIUploadInventoryAvailabilityStatusOutputRepresentation
getInventoryAvailabilityUploadStatus(String uploadId)
```

Parameters

`uploadId`
Type: `String`
The upload ID of the upload job.

Return Value
Type: `ConnectApi.OCIUploadInventoryAvailabilityStatusOutputRepresentation`

```java
getPublishLocationStructureStatus(uploadId)
```

Retrieve the status of a publish location structure job.

API Version
5.0
Requires Chatter
No

Signature

public static ConnectApi.OCIPublishLocationStructureStatusOutputRepresentation
getPublishLocationStructureStatus(String uploadId)

Parameters

uploadId
  Type: String
  The upload ID of the publish job.

Return Value

Type: ConnectApi.OCIPublishLocationStructureStatusOutputRepresentation

publishLocationStructure()

Asynchronously publish information about your inventory locations and location groups to Omnichannel Inventory. The publish includes records whose isEnabled and ShouldSyncWithOci fields are both true. This method returns an ID that you can use to retrieve the status of the publish job.

API Version
S1.0

Requires Chatter
No

Signature

public static ConnectApi.OCIPublishLocationStructureOutputRepresentation
publishLocationStructure()

Return Value

Type: ConnectApi.OCIPublishLocationStructureOutputRepresentation

releaseReservation(releaseReservationInputRepresentation)

Release one or more existing inventory reservations to free up that inventory.

API Version
S1.0
Requires Chatter
No

Signature
public static ConnectApi.OCIReleaseReservationOutputRepresentation
releaseReservation(ConnectApi.OCIReleaseReservationInputRepresentation
releaseReservationInputRepresentation)

Parameters
releaseReservationInputRepresentation
  Type: ConnectApi.OCIReleaseReservationInputRepresentation
  Details of one or more inventory reservations to release.

Return Value
Type: ConnectApi.OCIReleaseReservationOutputRepresentation

submitInventoryAvailabilityUpload(fileUpload)
Upload an inventory availability data file to Omnichannel Inventory.

API Version
51.0

Requires Chatter
No

Signature
public static ConnectApi.OCIUploadInventoryAvailabilityOutputRepresentation
submitInventoryAvailabilityUpload(ConnectApi.BinaryInput fileUpload)

Parameters
fileUpload
  Type: ConnectApi.BinaryInput
  JSON file containing inventory availability data.

Return Value
Type: ConnectApi.OCIUploadInventoryAvailabilityOutputRepresentation

Usage
To create an inventory data file, format the data as a series of JSON entries representing locations and individual inventory records.
Inventory Import Data Considerations:

- Separate the top-level entries with line feeds, not commas. Each entry must be on a single line.
- When the system reads a location entry, it assigns the subsequent inventory entries to that location until it reads another location entry.
- For each location entry, specify "mode": "UPDATE".
- Each inventory record entry requires a unique recordId. Best practice is to use a UUID. The recordId protects against importing duplicate data.
- Each inventory record entry requires an effectiveDate.
- Each futures entry requires a nonzero quantity and a future expectedDate.
- The system only creates or updates entries that are included in the data. It doesn’t delete entries that aren’t included. However, if an included entry has an empty value, that value is deleted.

Note: The file size limit is 10 MB, and it must be in JSON format. For larger data sets, you can use the headless Commerce API or split the data into multiple files. The headless Commerce API accepts GZIP or JSON files up to 100 MB in size.

This example illustrates the data format:

Note: For readability, this example shows the first few entries on multiple lines. In the import file, each location and inventory record entry must be on a single line.

```json
{
"location":"Warehouse-A", // location identifier
"mode":"UPDATE" // must be UPDATE (other operations might be available in future releases)
}
{
"recordId":"0a87539d-f3dd-47bc-91c7-9c752e39dbe0", // unique identifier for the inventory record
"onHand":10,
"sku":"12389156",
"effectiveDate":"2020-12-08T14:05:22.790896-07:00",
"futures": [ // list of future restocks
{ "quantity":1,
 "expectedDate":"2021-04-18T14:05:22.781-07:00"
 },
 { "quantity":5,
 "expectedDate":"2021-05-18T14:05:22.781-07:00"
 }
],
"safetyStockCount":0
}
{
"recordId":"0a87539d-f3dd-47bc-91c7-9c752e312345",
"onHand":10,
"sku":"9485728",
"effectiveDate":"2020-12-08T14:05:22.790896-07:00",
"futures": [ // list of future restocks
{ "quantity":10,
 "expectedDate":"2021-04-18T14:05:22.781-07:00"
 }
]
```
transferReservation(transferReservationInputRepresentation)

Transfer one or more inventory reservations between locations or location groups. This API doesn't change physical quantities, but reduces the reserved quantity at the source and increases it at the destination.

API Version
51.0

Requires Chatter
No

Signature
public static ConnectApi.OCITransferReservationOutputRepresentation transferReservation(ConnectApi.OCITransferReservationInputRepresentation transferReservationInputRepresentation)

Parameters

transferReservationInputRepresentation
Type: ConnectApi.OCITransferReservationInputRepresentation
Wraps a list of inventory reservation transfers and specifies whether a single failure cancels the entire list.

Return Value
Type: ConnectApi.OCITransferReservationOutputRepresentation

Orchestration Class
Get orchestration instances.

Namespace
ConnectApi

Orchestration Methods
The following are methods for Orchestration. All methods are static.
IN THIS SECTION:

getOrchestrationInstanceCollection(relatedRecordId)
Get orchestration instances associated with a Salesforce record that’s configured as a context record for orchestration interactive steps.

getOrchestrationInstanceCollection(relatedRecordId)
Get orchestration instances associated with a Salesforce record that’s configured as a context record for orchestration interactive steps.

API Version
54.0

Requires Chatter
No

Signature
public static ConnectApi.OrchestrationInstanceCollection getOrchestrationInstanceCollection(String relatedRecordId)

Parameters
relatedRecordId
Type: String
The ID of a record configured as a context record for orchestration interactive steps.

Return Value
Type: ConnectApi.OrchestrationInstanceCollection

OrderPaymentSummary Class
Work with payments in Order Management.

Namespace
ConnectApi

OrderPaymentSummary Methods
The following are methods for OrderPaymentSummary. All methods are static.
IN THIS SECTION:

createOrderPaymentSummary(orderPaymentSummaryInput)

Create an OrderPaymentSummary for an OrderSummary. Specify a payment authorization or payments that share the same payment method. In an org with the multicurrency feature enabled, the OrderPaymentSummary inherits the CurrencyIsoCode value from the OrderSummary.

createOrderPaymentSummary(orderPaymentSummaryInput)

Create an OrderPaymentSummary for an OrderSummary. Specify a payment authorization or payments that share the same payment method. In an org with the multicurrency feature enabled, the OrderPaymentSummary inherits the CurrencyIsoCode value from the OrderSummary.

API Version

48.0

Requires Chatter

No

Signature

public static ConnectApi.CreateOrderPaymentSummaryOutputRepresentation createOrderPaymentSummary(ConnectApi.CreateOrderPaymentSummaryInputRepresentation orderPaymentSummaryInput)

Parameters

orderPaymentSummaryInput

Type: ConnectApi.CreateOrderPaymentSummaryInputRepresentation

The OrderSummary and payment authorization or payments.

Return Value

Type: ConnectApi.CreateOrderPaymentSummaryOutputRepresentation

Example

String orderSummaryId = '1Osxx0000004CCG';
String paymentId1 = '0a3xx0000000085AAA';
String paymentId2 = '0a3xx0000000085BBB';

ConnectApi.CreateOrderPaymentSummaryInputRepresentation orderPaymentSummaryInput = new
ConnectApi.CreateOrderPaymentSummaryInputRepresentation();
orderPaymentSummaryInput.orderSummaryId = orderSummaryId;
List<String> paymentList = new List<String>();
paymentList.add(paymentId1);
paymentList.add(paymentId2);
orderPaymentSummaryInput.paymentIds = paymentList;
ConnectApi.CreateOrderPaymentSummaryOutputRepresentation result =
  ConnectAPI.OrderPaymentSummary.createOrderPaymentSummary(orderPaymentSummaryInput);

OrderSummary Class

Work with orders in Order Management.

Namespace

ConnectApi

OrderSummary Methods

The following are methods for OrderSummary. All methods are static.

IN THIS SECTION:

  adjustPreview(orderSummaryId, adjustInput)
  Retrieve the expected results of adjusting the price of one or more OrderItemSummaries from an OrderSummary, without actually
  executing the adjustment. The response data contains the financial changes that would result from submitting the proposed
  adjustment.

  adjustSubmit(orderSummaryId, adjustInput)
  Adjust the price of one or more OrderItemSummaries from an OrderSummary, and create corresponding change orders.

  createCreditMemo(orderSummaryId, creditMemoInput)
  Create a credit memo to represent the refund for one or more change orders associated with an OrderSummary.

  createMultipleInvoices(invoicesInput)
  Create Invoices to represent the charges for one or more change orders. Create Invoices for change orders that increase order
  amounts, such as for return fees. When you ensure the refund for a return, include the invoices for any associated return fees in the
  request.

  ensureFundsAsync(orderSummaryId, ensureFundsInput)
  Ensure funds for an Invoice and apply them to it. If needed, capture authorized funds by sending a request to a payment provider.
  This method inserts a background operation into an asynchronous job queue and returns the ID of that operation so you can track
  its status. Payment gateway responses appear in the payment gateway log and do not affect the background operation status.

  ensureRefundsAsync(orderSummaryId, ensureRefundsInput)
  Ensure refunds for a CreditMemo or excess funds by sending a request to a payment provider. This method inserts a background
  operation into an asynchronous job queue and returns the ID of that operation so you can track its status. Payment gateway responses
  appear in the payment gateway log and do not affect the background operation status.

  multipleEnsureFundsAsync(multipleEnsureFundsInput)
  Ensure and apply funds for one or more Invoices. If needed, capture authorized funds by sending a request to a payment provider.
  This method inserts a background operation into an asynchronous job queue and returns the ID of that operation so you can track
  its status. Payment gateway responses appear in the payment gateway log and do not affect the background operation status.

  previewCancel(orderSummaryId, changeInput)
  Retrieve the expected change order values for canceling one or more OrderItemSummaries from an OrderSummary, without actually
  executing the cancel.
previewReturn(orderSummaryId, changelInput)
Retrieve the expected change order values for a simple return of one or more OrderItemSummaries from an OrderSummary, without actually executing the return.

submitCancel(orderSummaryId, changelInput)
Cancel one or more OrderItemSummaries from an OrderSummary, and create a corresponding change order.

submitReturn(orderSummaryId, changelInput)
Return one or more OrderItemSummaries from an OrderSummary, and create a corresponding change order. This return is a simple return that creates a change order but not a ReturnOrder.

adjustPreview(orderSummaryId, adjustInput)
Retrieve the expected results of adjusting the price of one or more OrderItemSummaries from an OrderSummary, without actually executing the adjustment. The response data contains the financial changes that would result from submitting the proposed adjustment.

API Version
49.0

Requires Chatter
No

Signature
public static ConnectApi.AdjustOrderSummaryOutputRepresentation adjustPreview(String orderSummaryId, ConnectApi.AdjustOrderItemSummaryInputRepresentation adjustInput)

Parameters
orderSummaryId
Type: String
ID of the OrderSummary.

adjustInput
Type: ConnectApi.AdjustOrderItemSummaryInputRepresentation
Price adjustments to OrderItemSummaries that together make up a price adjustment to an order, with options for adjusting items in the process of being fulfilled.

Return Value
Type: ConnectApi.AdjustOrderSummaryOutputRepresentation

Usage
When a price adjustment is applied to an OrderItemSummary, its quantities are considered in three groups:

Pre-fulfillment
QuantityAvailableToFulfill, which is equal to QuantityOrdered - QuantityCanceled - QuantityAllocated
In-fulfillment
  QuantityAllocated - QuantityFulfilled

Post-fulfillment
  QuantityAvailableToReturn, which is equal to QuantityFulfilled - QuantityReturnInitiated

You can apply adjustments to these groups in three different ways, controlled by the allocatedItemsChangeOrderType input property:

- Distribute the adjustment evenly between pre-fulfillment and post-fulfillment quantities. Ignore in-fulfillment quantities. Submitting the adjustment would create one change order for the adjustments to pre-fulfillment quantities and one change order for the adjustments to post-fulfillment quantities.
- Distribute the adjustment evenly between pre-fulfillment, in-fulfillment, and post-fulfillment quantities. Submitting the adjustment would create one change order for the adjustments to both pre-fulfillment and in-fulfillment quantities, and one change order for the adjustments to post-fulfillment quantities.
- Distribute the adjustment evenly between pre-fulfillment, in-fulfillment, and post-fulfillment quantities. Submitting the adjustment would create one change order for the adjustments to pre-fulfillment quantities, one change order for the adjustments to in-fulfillment quantities, and one change order for the adjustments to post-fulfillment quantities.

SEE ALSO:
  createCreditMemo(orderSummaryId, creditMemoInput)
  ensureRefundsAsync(orderSummaryId, ensureRefundsInput)
  adjustSubmit(orderSummaryId, adjustInput)

adjustSubmit(orderSummaryId, adjustInput)

Adjust the price of one or more OrderItemSummaries from an OrderSummary, and create corresponding change orders.

API Version
49.0

Requires Chatter
No

Signature
public static ConnectApi.AdjustOrderSummaryOutputRepresentation adjustSubmit(String orderSummaryId, ConnectApi.AdjustOrderItemSummaryInputRepresentation adjustInput)

Parameters
  orderSummaryId
    Type: String
    ID of the OrderSummary.
  adjustInput
    Type: ConnectApi.AdjustOrderItemSummaryInputRepresentation
Price adjustments to OrderItemSummaries that together make up a price adjustment to an order, with options for adjusting items in the process of being fulfilled.

Return Value
Type: `ConnectApi.AdjustOrderSummaryOutputRepresentation`

Usage
When a price adjustment is applied to an OrderItemSummary, its quantities are considered in three groups:

**Pre-fulfillment**
QuantityAvailableToFulfill, which is equal to QuantityOrdered - QuantityCanceled - QuantityAllocated

**In-fulfillment**
QuantityAllocated - QuantityFulfilled

**Post-fulfillment**
QuantityAvailableToReturn, which is equal to QuantityFulfilled - QuantityReturnInitiated

You can apply adjustments to these groups in three different ways, controlled by the `allocatedItemsChangeOrderType` input property:

- Distribute the adjustment evenly between pre-fulfillment and post-fulfillment quantities. Ignore in-fulfillment quantities. Create one change order for the adjustments to pre-fulfillment quantities and one change order for the adjustments to post-fulfillment quantities.
- Distribute the adjustment evenly between pre-fulfillment, in-fulfillment, and post-fulfillment quantities. Create one change order for the adjustments to both pre-fulfillment and in-fulfillment quantities, and one change order for the adjustments to post-fulfillment quantities.
- Distribute the adjustment evenly between pre-fulfillment, in-fulfillment, and post-fulfillment quantities. Create one change order for the adjustments to pre-fulfillment quantities, one change order for the adjustments to in-fulfillment quantities, and one change order for the adjustments to post-fulfillment quantities.

After submitting a price adjustment, process refunds as appropriate:

- If the discount only applied to OrderItemSummaries for which payment hasn’t been captured, it doesn’t require a refund. This situation normally applies to OrderItemSummaries in the US that haven’t been fulfilled.
- If the discount applied to OrderItemSummaries that haven’t been fulfilled and for which payment has been captured, process a refund. In this case, pass the `totalExcessFundsAmount` from the output representation to the `ensureRefundsAsync()` method.
- If the discount applied to OrderItemSummaries that have been fulfilled, process a refund. Pass the `postFulfillmentChangeOrderId` from the output representation to the `createCreditMemo()` method, then pass the CreditMemo to the `ensureRefundsAsync()` method.
- If the discount applied to both fulfilled and unfulfilled OrderItemSummaries for which payment has been captured, process both refunds. Pass the `postFulfillmentChangeOrderId` from the output representation to the `createCreditMemo()` method, then pass the credit memo and the `totalExcessFundsAmount` from the output representation to the `ensureRefundsAsync()` method.

SEE ALSO:
- `createCreditMemo(orderSummaryId, creditMemoInput)`
- `ensureRefundsAsync(orderSummaryId, ensureRefundsInput)`
- `adjustPreview(orderSummaryId, adjustInput)`
createCreditMemo(orderSummaryId, creditMemoInput)
Create a credit memo to represent the refund for one or more change orders associated with an OrderSummary.

API Version
48.0

Requires Chatter
No

Signature
public static ConnectApi.CreateCreditMemoOutputRepresentation createCreditMemo(String orderSummaryId, ConnectApi.CreateCreditMemoInputRepresentation creditMemoInput)

Parameters
orderSummaryId
Type: String
ID of the OrderSummary.

creditMemoInput
Type: ConnectApi.CreateCreditMemoInputRepresentation
The list of change order IDs.

Return Value
Type: ConnectApi.CreateCreditMemoOutputRepresentation

createMultipleInvoices(invoicesInput)
Create Invoices to represent the charges for one or more change orders. Create Invoices for change orders that increase order amounts, such as for return fees. When you ensure the refund for a return, include the invoices for any associated return fees in the request.

API Version
56.0

Requires Chatter
No

Signature
public static ConnectApi.CreateMultipleInvoicesFromChangeOrdersOutputRepresentation createMultipleInvoices(ConnectApi.CreateMultipleInvoicesFromChangeOrdersInputRepresentation invoicesInput)
Parameters

`invoicesInput`
- **Type:** `ConnectApi.CreateMultipleInvoicesFromChangeOrdersInputRepresentation`
  - Data about the change orders to create Invoices for.

Return Value

- **Type:** `ConnectApi.CreateMultipleInvoicesFromChangeOrdersOutputRepresentation`

SEE ALSO:
- `ensureRefundsAsync(orderSummaryId, ensureRefundsInput)`
- `createReturnOrder(returnOrderInput)`
- `returnItems(returnOrderId, returnItemsInput)`

`ensureFundsAsync(orderSummaryId, ensureFundsInput)`

Ensure funds for an Invoice and apply them to it. If needed, capture authorized funds by sending a request to a payment provider. This method inserts a background operation into an asynchronous job queue and returns the ID of that operation so you can track its status. Payment gateway responses appear in the payment gateway log and do not affect the background operation status.

**API Version**

- 48.0

**Requires Chatter**

- No

**Signature**

```java
public static ConnectApi.EnsureFundsAsyncOutputRepresentation ensureFundsAsync(String orderSummaryId, ConnectApi.EnsureFundsAsyncInputRepresentation ensureFundsInput)
```

Parameters

- **orderSummaryId**
  - **Type:** `String`
  - ID of the OrderSummary.

- **ensureFundsInput**
  - **Type:** `ConnectApi.EnsureFundsAsyncInputRepresentation`
  - The ID of the Invoice.

Return Value

- **Type:** `ConnectApi.EnsureFundsAsyncOutputRepresentation`
Usage
This method checks the OrderPaymentSummaries associated with the specified OrderSummary for funds to apply to the Invoice balance following this logic:

- **Note:** If multiple OrderPaymentSummaries have equal BalanceAmount values, their order of selection is random.

1. Verify that the Invoice balance doesn't exceed the total BalanceAmount of all the OrderPaymentSummaries associated with the OrderSummary.
2. If an OrderPaymentSummary has a BalanceAmount equal to the Invoice balance, apply the funds from that OrderPaymentSummary.
3. If no exact match was found, apply funds from the OrderPaymentSummary with the largest BalanceAmount.
4. If the Invoice still has a balance to ensure, repeat steps 2 and 3 until the full balance is ensured or no captured funds remain.
5. If the Invoice still has a balance, look for an OrderPaymentSummary with an authorized amount equal to the remaining Invoice balance. If one exists, capture and apply the funds from that OrderPaymentSummary.
6. If no exact match was found, capture and apply funds from the OrderPaymentSummary with the largest authorized amount.
7. If the Invoice still has a balance to ensure, repeat steps 5 and 6 until the full balance is ensured.

- **Note:** If the method creates a payment, the payment record's ClientContext value isn't predictable. Don't use it in custom logic.

SEE ALSO:
- `multipleEnsureFundsAsync(multipleEnsureFundsInput)`
- `ensureRefundsAsync(orderSummaryId, ensureRefundsInput)`

**ensureRefundsAsync(orderSummaryId, ensureRefundsInput)**
Ensure refunds for a CreditMemo or excess funds by sending a request to a payment provider. This method inserts a background operation into an asynchronous job queue and returns the ID of that operation so you can track its status. Payment gateway responses appear in the payment gateway log and do not affect the background operation status.

**API Version**
48.0

**Requires Chatter**
No

**Signature**
```java
public static ConnectApi.EnsureRefundsAsyncOutputRepresentation ensureRefundsAsync(String orderSummaryId, ConnectApi.EnsureRefundsAsyncInputRepresentation ensureRefundsInput)
```

**Parameters**
- `orderId`  
  Type: String  
  ID of the OrderSummary.
**ensureRefundsInput**
Type: `ConnectApi.EnsureRefundsAsyncInputRepresentation`

ID of a credit memo to ensure refunds for, an amount of excess funds to refund, or both. At least one is required. Also includes any invoices for fees that reduce the refund amount, such as return fees. If multiple payment methods are available, you can specify how to distribute the refund.

**Return Value**
Type: `ConnectApi.EnsureRefundsAsyncOutputRepresentation`

**Usage**
This method applies the refund to the OrderPaymentSummaries associated with the specified OrderSummary following this logic.

- **Note:** If multiple OrderPaymentSummaries have equal AvailableToRefund amounts, their order of selection is random.

1. Verify that the CreditMemo balance and excess funds amount don’t exceed the total AvailableToRefund amount of all the OrderPaymentSummaries associated with the OrderSummary.

2. If `sequences` is specified, follow these steps.
   - a. Traverse the `sequences` list in order and apply the specified refund amounts to the specified OrderPaymentSummaries.
   - b. If the specified CreditMemo and excess funds are fully refunded, or if `isAllowPartial` is true, then the action stops here.

3. If a CreditMemo is specified, follow these steps.
   - a. If an OrderPaymentSummary has an AvailableToRefund amount matching the CreditMemo’s remaining balance, apply the refund to that payment.
   - b. If no exact match was found, apply the refund to the OrderPaymentSummary with the largest AvailableToRefund amount.
   - c. If the CreditMemo has any remaining balance, repeat steps a and b until that balance is fully refunded.

4. If an excess funds amount is specified, follow these steps.
   - a. Examine those OrderPaymentSummaries. If one has an AvailableToRefund amount matching the excess funds amount, apply the refund to that payment.
   - b. If no exact match was found, apply the refund to the OrderPaymentSummary with the largest AvailableToRefund amount.
   - c. If any excess funds amount remains, repeat steps a and b until it’s fully refunded.

- **Note:** If the method creates a refund, the refund record’s ClientContext value isn’t predictable. Don’t use it in custom logic.

**SEE ALSO:**
- `createReturnOrder(returnOrderInput)`
- `returnItems(returnOrderId, returnItemsInput)`
- `createMultipleInvoices(invoicesInput)`
**multipleEnsureFundsAsync(multipleEnsureFundsInput)**

Ensure and apply funds for one or more Invoices. If needed, capture authorized funds by sending a request to a payment provider. This method inserts a background operation into an asynchronous job queue and returns the ID of that operation so you can track its status. Payment gateway responses appear in the payment gateway log and do not affect the background operation status.

**API Version**

56.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.MultipleAsyncOutputRepresentation multipleEnsureFundsAsync(ConnectApi.MultipleEnsureFundsAsyncInputRepresentation multipleEnsureFundsInput)
```

**Parameters**

`multipleEnsureFundsInput`

- **Type**: `ConnectApi.MultipleEnsureFundsAsyncInputRepresentation`
  - List of Invoices and the associated OrderSummaries.

**Return Value**

- **Type**: `ConnectApi.MultipleAsyncOutputRepresentation`

**Usage**

For each Invoice in the request, this method checks the OrderPaymentSummaries associated with the specified OrderSummary for funds to apply to the Invoice balance following this logic.

**Note**: If multiple OrderPaymentSummaries have equal `BalanceAmount` values, their order of selection is random.

1. Verify that the Invoice balance doesn’t exceed the total `BalanceAmount` of all the OrderPaymentSummaries associated with the OrderSummary.
2. If an OrderPaymentSummary has a `BalanceAmount` equal to the invoice balance, apply the funds from that OrderPaymentSummary.
3. If no exact match was found, apply funds from the OrderPaymentSummary with the largest `BalanceAmount`.
4. If the Invoice still has a balance to ensure, repeat steps 2 and 3 until the full balance is ensured or no captured funds remain.
5. If the Invoice still has a balance, look for an OrderPaymentSummary with an authorized amount equal to the remaining Invoice balance. If one exists, capture and apply the funds from that OrderPaymentSummary.
6. If no exact match was found, capture and apply funds from the OrderPaymentSummary with the largest authorized amount.
7. If the Invoice still has a balance to ensure, repeat steps 5 and 6 until the full balance is ensured.
Note: If the method creates a payment, the payment record’s ClientContext value isn’t predictable. Don’t use it in custom logic.

SEE ALSO:
  ensureFundsAsync(orderSummaryId, ensureFundsInput)

previewCancel(orderSummaryId, changeInput)
Retrieve the expected change order values for canceling one or more OrderItemSummaries from an OrderSummary, without actually executing the cancel.

API Version
48.0

Requires Chatter
No

Signature
public static ConnectApi.PreviewCancelOutputRepresentation previewCancel(String orderSummaryId, ConnectApi.ChangeInputRepresentation changeInput)

Parameters
orderSummaryId
  Type: String
  ID of the OrderSummary.

crateInput
  Type: ConnectApi.ChangeInputRepresentation
  A list of changes to OrderItemSummaries that make up an order change, such as a cancel or return.

Return Value
Type: ConnectApi.PreviewCancelOutputRepresentation

SEE ALSO:
  createCreditMemo(orderSummaryId, creditMemoInput)
  ensureRefundsAsync(orderSummaryId, ensureRefundsInput)
  submitCancel(orderSummaryId, changeInput)

previewReturn(orderSummaryId, changeInput)
Retrieve the expected change order values for a simple return of one or more OrderItemSummaries from an OrderSummary, without actually executing the return.
API Version
48.0

Requires Chatter
No

Signature
public static ConnectApi.PreviewReturnOutputRepresentation previewReturn(String orderSummaryId, ConnectApi.ChangeInputRepresentation changeInput)

Parameters
orderSummaryId
   Type: String
   ID of the OrderSummary.
changeInput
   Type: ConnectApi.ChangeInputRepresentation
   A list of changes to OrderItemSummaries that make up an order change, such as a cancel or return.

Return Value
Type: ConnectApi.PreviewReturnOutputRepresentation

SEE ALSO:
   createCreditMemo(orderSummaryId, creditMemoInput)
   ensureRefundsAsync(orderSummaryId, ensureRefundsInput)
   submitReturn(orderSummaryId, changeInput)

submitCancel(orderSummaryId, changeInput)
Cancel one or more OrderItemSummaries from an OrderSummary, and create a corresponding change order.

API Version
48.0

Requires Chatter
No

Signature
public static ConnectApi.SubmitCancelOutputRepresentation submitCancel(String orderSummaryId, ConnectApi.ChangeInputRepresentation changeInput)
Parameters

orderSummaryId
Type: String
ID of the OrderSummary.

changeInput
Type: ConnectApi.ChangeInputRepresentation
A list of changes to OrderItemSummaries that make up an order change, such as a cancel or return.

Return Value
Type: ConnectApi.SubmitCancelOutputRepresentation

SEE ALSO:
createCreditMemo(orderSummaryId, creditMemoInput)
ensureRefundsAsync(orderSummaryId, ensureRefundsInput)
previewCancel(orderSummaryId, changeInput)

submitReturn(orderSummaryId, changeInput)
Return one or more OrderItemSummaries from an OrderSummary, and create a corresponding change order. This return is a simple return that creates a change order but not a ReturnOrder.

API Version
48.0

Requires Chatter
No

Signature
public static ConnectApi.SubmitReturnOutputRepresentation submitReturn(String orderSummaryId, ConnectApi.ChangeInputRepresentation changeInput)

Parameters

orderSummaryId
Type: String
ID of the OrderSummary.

changeInput
Type: ConnectApi.ChangeInputRepresentation
A list of changes to OrderItemSummaries that make up an order change, such as a cancel or return.

Return Value
Type: ConnectApi.SubmitReturnOutputRepresentation
Usage

After submitting a return, process a refund. Pass the `changeOrderId` from the output representation to the `createCreditMemo()` method, then pass the credit memo to the `ensureRefundsAsync()` method.

SEE ALSO:
- `createCreditMemo(orderSummaryId, creditMemoInput)`
- `ensureRefundsAsync(orderSummaryId, ensureRefundsInput)`
- `previewReturn(orderSummaryId, changeInput)`

OrderSummaryCreation Class

Create Order Summaries in Order Management.

Namespace

ConnectApi

OrderSummaryCreation Methods

The following are methods for `OrderSummaryCreation`. All methods are static.

IN THIS SECTION:
- `createOrderSummary(orderSummaryInput)`

`createOrderSummary(orderSummaryInput)`

Create an OrderSummary based on an order. That order is considered the original order for the OrderSummary. Subsequent change orders that apply to the OrderSummary are also represented as orders. You can specify whether the order is managed in Salesforce Order Management or by an external system. Most Salesforce Order Management APIs can run only on orders that it manages.

API Version

48.0

Requires Chatter

No

Signature

```java
public static ConnectApi.OrderSummaryOutputRepresentation createOrderSummary(ConnectApi.OrderSummaryInputRepresentation orderSummaryInput)
```
Parameters

orderSummaryInput
Type: ConnectApi.OrderSummaryInputRepresenation
Input object that wraps the ID of the source order.

Return Value
Type: ConnectApi.OrderSummaryOutputRepresentation

Organization Class
Access information about an org.

Namespace
ConnectApi

Organization Methods
The following are methods for Organization. All methods are static.

IN THIS SECTION:

getSettings()
Get information about the context user and the org, including which features are enabled.

getSettings()
Get information about the context user and the org, including which features are enabled.

API Version
28.0

Requires Chatter
No

Signature
public static ConnectApi. OrganizationSettings getSettings()

Return Value
Type: ConnectApi.OrganizationSettings

PardotBusinessUnitContext Class
Get the Pardot business units the context user has access to.
Namespace

ConnectApi

PardotBusinessUnitContext Methods

The following are methods for PardotBusinessUnitContext. All methods are static.

IN THIS SECTION:

- `getBusinessUnitContext()`: Get the Pardot business units the context user has access to.
- `getBusinessUnitContextByIsCurrentStatus(isCurrent)`: Get the Pardot business units the context user has access to by specifying the current status.

`getBusinessUnitContext()`

Get the Pardot business units the context user has access to.

API Version

55.0

Requires Chatter

No

Signature

`public static ConnectApi.PardotBusinessUnitContextOutput getBusinessUnitContext()`

Return Value

Type: `ConnectApi.PardotBusinessUnitContextOutput`

`getBusinessUnitContextByIsCurrentStatus(isCurrent)`

Get the Pardot business units the context user has access to by specifying the current status.

API Version

55.0

Requires Chatter

No
Signature

```java
public static ConnectApi.PardotBusinessUnitContextOutput
getBusinessUnitContextByIsCurrentStatus(Boolean isCurrent)
```

Parameters

```java
isCurrent
Type: Boolean
```
Specifies whether to return only the business unit that’s selected as the context user’s current business unit context in the business unit switcher of the Pardot Lightning app (true) or to return only the business units that aren’t selected as the context user’s current business unit context (false).

Return Value

Type: `ConnectApi.PardotBusinessUnitContextOutput`

Payments Class

Authorize a payment, capture an authorized payment, and refund an authorized payment.

Namespace

`ConnectApi`

Payments Methods

The following are methods for `Payments`. All methods are static.

IN THIS SECTION:

```java
authorize(authorizePayment)
Authorize a payment.

postAuth(postAuthorizePayment)
Confirms that the merchant is ready to capture payment of an existing pre-authorized transaction.

reverseAuthorization(AuthReversalInput, authorizationId)
Reverses a payment authorization.

capture(AuthCaptureInput, authorizationId)
Capture an authorized payment.

refund(ReferencedRefundInput, paymentId)
Refund an authorized payment.

sale(sale)
Captures a payment without any prior authorization and creates a payment entity. The payment sale transaction is a combination of an `Authorize` transaction and `Capture` transaction. This payment sale method allows merchants to request that the funds are transferred to the merchant account in a single command, with no further action (such as charging a credit card) from the merchant.
```
**tokenizePaymentMethod(tokenizePaymentMethodInput)**

Method to take the input parameters of the payment method you want to tokenize and then pass them to the payment gateway’s tokenization service. The results of the tokenization request are returned as a response from the payment gateway.

**authorize(authorizePayment)**

Authorize a payment.

**API Version**

51.0

**Requires Chatter**

No

**Signature**

```
global static ConnectApi.AuthorizationResponse authorize(ConnectApi.AuthorizationRequest authorizePayment)
```

**Parameters**

`authorizePayment`

Type: `ConnectApi.AuthorizationRequest`

Represents a payment authorization.

**Return Value**

Type: `ConnectApi.AuthorizationResponse`

**postAuth(postAuthorizePayment)**

Confirms that the merchant is ready to capture payment of an existing pre-authorized transaction.

**API Version**

54.0

**Requires Chatter**

No

**Signature**

```
global static ConnectApi.PostAuthorizationResponse postAuth(ConnectApi.PostAuthRequest postAuthorizePayment)
```
Parameters

`postAuthorizePayment`
- Type: `ConnectApi.PostAuthRequest`
  - Information about the payment, payment method, and payment gateway from the original payment authorization.

Return Value

Type: `ConnectApi.PostAuthorizationResponse`

`reverseAuthorization(AuthReversalInput, authorizationId)`
Reverses a payment authorization.

API Version

51.0

Requires Chatter

No

Signature

```java
```

Parameters

`AuthReversalInput`
- Type: `ConnectApi.AuthorizationReversalRequest`
  - Input information for the payment authorization reversal.

`authorizationId`
- Type: `String`
  - The ID of the payment authorization to be reversed.

Return Value

Type: `ConnectApi.AuthorizationReversalResponse`

`capture(AuthCaptureInput, authorizationId)`
Capture an authorized payment.

To access Payments methods, you need these permissions.

- Salesforce Order Management License
- PaymentsAPIUser user permission. This permission is available with the Salesforce Order Management License. Your Salesforce admin assigns it to your profile.
API Version
50.0

Requires Chatter
No

Signature
global static ConnectApi.CaptureResponse capture(ConnectApi.CaptureRequest AuthCaptureInput, String authorizationId)

Parameters
AuthCaptureInput
Type: ConnectApi.CaptureRequest
A ConnectApi.CaptureRequest object with information about the payment capture.

authorizationId
Type: String
ID of the payment authorization. Required.

Return Value
Type: ConnectApi.CaptureResponse

refund(ReferencedRefundInput, paymentId)
Refund an authorized payment.

To access Payments methods, you need these permissions.
- Salesforce Order Management License
- PaymentsAPIUser user permission. This permission is available with the Salesforce Order Management License. Your Salesforce admin assigns it to your profile.

API Version
50.0

Requires Chatter
No

Signature
global static ConnectApi.ReferencedRefundResponse refund(ConnectApi.ReferencedRefundRequest ReferencedRefundInput, String paymentId)
Parameters

ReferencedRefundInput

Type: ConnectApi.ReferencedRefundRequest

A ConnectApi.ReferencedRefundRequest object with information about the refund.

paymentId

Type: String

ID of the payment to be refunded. Required.

Return Value

Type: ConnectApi.ReferencedRefundResponse

sale(sale)

Captures a payment without any prior authorization and creates a payment entity. The payment sale transaction is a combination of an Authorize transaction and Capture transaction. This payment sale method allows merchants to request that the funds are transferred to the merchant account in a single command, with no further action (such as charging a credit card) from the merchant.

API Version

S4.0

Requires Chatter

No

Signature

global static ConnectApi.SaleResponse sale(ConnectApi.SaleRequest sale)

Parameters

sale

Type: ConnectApi.SaleRequest

Payment sale input class.

Return Value

Type: ConnectApi.SaleResponse

tokenizePaymentMethod(tokenizePaymentMethodInput)

Method to take the input parameters of the payment method you want to tokenize and then pass them to the payment gateway's tokenization service. The results of the tokenization request are returned as a response from the payment gateway.

API Version

S2.0
Requires Chatter

No

Signature

global static ConnectApi.PaymentMethodTokenizationResponse tokenizePaymentMethod(ConnectApi.PaymentMethodTokenizationRequest tokenizePaymentMethodInput)

Parameters

tokenizePaymentMethodInput
Type: ConnectApi.PaymentMethodTokenizationRequest
Information about the payment method to be tokenized.

Return Value
Type: ConnectApi.PaymentMethodTokenizationResponse

Usage
Accepts input parameters representing a payment method and passes them in a tokenization request to the payment gateway. The results of the tokenization request are returned as a response from the payment gateway. If the tokenization was successful, the response contains the tokenized value and details about the tokenization process. Otherwise, the response contains an error message and details about the error.

Example

ConnectApi.PaymentMethodTokenizationRequest request = new 
ConnectApi.PaymentMethodTokenizationRequest();
request.paymentGatewayId = '0b0xx0000001Ja5AAE';
ConnectApi.CardPaymentMethodRequest cpmRequest = new ConnectApi.CardPaymentMethodRequest();
cpmRequest.cardHolderName = 'Jo Manager';
cpmRequest.expiryMonth = 11;
cpmRequest.expiryYear = 2222;
cpmRequest.cardNumber = '4111111111111111';
cpmRequest.cvv = '111';
cpmRequest.cardCategory = ConnectApi.CardCategory.CreditCard;
cpmRequest.cardType = ConnectApi.CardType.Visa.name();
request.cardPaymentMethod = cpmRequest;
ConnectApi.PaymentMethodTokenizationResponse response =
ConnectApi.Payments.tokenizePaymentMethod(request);

Personalization Class

Get assigned personalization audiences that match the user context. Create, get, update, and delete an audience. Get personalization targets that match the user context, based on the assigned audiences that include the user. Create and update targets. Get and delete a target.
Namespace

ConnectApi

Note: Personalization varies what the user can see in the browser but doesn’t secure data in any way. To prevent users accessing sensitive data, use standard Salesforce security features, such as sharing rules and permission sets.

Personalization Methods

The following are methods for Personalization. All methods are static.

IN THIS SECTION:

createAudience(communityId, audience)
Create an audience.

createTargets(communityId, target)
Create targets.

deleteAudience(communityId, audienceId)
Delete an audience.

deleteTarget(communityId, targetId)
Delete a target.

getAudience(communityId, audienceId, includeAudienceCriteria)
Get an audience.

getAudienceBatch(communityId, audienceIds)
Get audience information for a comma-separated list of audience IDs.

getAudiences(communityId, ipAddress, domain, userId, publishStatus, includeAudienceCriteria, targetTypes, recordId)
Get a list of assigned audiences that match the user context and record information.

getAudiences(communityId, ipAddress, domain, userId, publishStatus, includeAudienceCriteria, targetTypes)
Get a list of assigned audiences that match the user context.

getTarget(communityId, targetId)
Get a target.

getTargetBatch(communityId, targetIds)
Get target information for a comma-separated list of target IDs.

getTargets(communityId, ipAddress, domain, userId, publishStatus, recordId, targetTypes, includeAudience, includeAllMatchingTargetsWithinGroup, groupNames)
Get a list of targets that match the user context, based on the assigned audiences that include the user.

updateAudience(communityId, audienceId, audience)
Update an audience.

updateTargets(communityId, target)
Update targets.

createAudience(communityId, audience)

Create an audience.
API Version
48.0

Requires Chatter
No

Signature
public static ConnectApi.Audience createAudience(String communityId,
ConnectApi.AudienceInput audience)

Parameters

communityId
Type: String
ID of the Experience Cloud site.

audience
Type: ConnectApi.AudienceInput
A ConnectApi.AudienceInput object that defines the audience.

Return Value
Type: ConnectApi.Audience

createTargets(communityId, target)
Create targets.

API Version
48.0

Requires Chatter
No

Signature
public static ConnectApi.TargetCollection createTargets(String communityId,
ConnectApi.TargetCollectionInput target)

Parameters

communityId
Type: String
ID of the Experience Cloud site.
target
   Type: ConnectApi.TargetCollectionInput
   A ConnectApi.TargetCollectionInput object that defines the targets.

Return Value
Type: ConnectApi.TargetCollection

deleteAudience(communityId, audienceId)
Delete an audience.

API Version
48.0

Requires Chatter
No

Signature
public static Void deleteAudience(String communityId, String audienceId)

Parameters
communityId
   Type: String
   ID of the Experience Cloud site.

audienceId
   Type: String
   ID of the audience.

Return Value
Type: Void

deleteTarget(communityId, targetId)
Delete a target.

API Version
48.0

Requires Chatter
No
Signature

```java
public static Void deleteTarget(String communityId, String targetId)
```

Parameters

- `communityId`
  - Type: `String`
  - ID of the Experience Cloud site.
- `targetId`
  - Type: `String`
  - ID of the target.

Return Value

Type: Void

---

```java
getAudience(communityId, audienceId, includeAudienceCriteria)
```

Get an audience.

**API Version**

48.0

- **Available to Guest Users**
  - 48.0
- **Requires Chatter**
  - No

Signature

```java
public static ConnectApi.Audience getAudience(String communityId, String audienceId, Boolean includeAudienceCriteria)
```

Parameters

- `communityId`
  - Type: `String`
  - ID of the Experience Cloud site.
- `audienceId`
  - Type: `String`
  - ID of the audience.
- `includeAudienceCriteria`
  - Type: `Boolean`
Specifies whether to include audience criteria (true) or not (false). If unspecified, defaults to false.

Return Value
Type: ConnectApi.Audience

getAudienceBatch(communityId, audienceIds)
Get audience information for a comma-separated list of audience IDs.

API Version
48.0

Available to Guest Users
48.0

Requires Chatter
No

Signature
public static ConnectApi.BatchResult[] getAudienceBatch(String communityId, List<String> audienceIds)

Parameters
communityId
Type: String
ID of the Experience Cloud site.
audienceIds
Type: List<String>
Comma-separated list of audience IDs.

Return Value
Type: ConnectApi.BatchResult[]
The ConnectApi.BatchResult.getResult() method returns a ConnectApi.Audience object and errors for audiences that didn’t load.

getAudiences(communityId, ipAddress, domain, userId, publishStatus, includeAudienceCriteria, targetTypes, recordId)
Get a list of assigned audiences that match the user context and record information.
API Version
51.0

Available to Guest Users
51.0

Requires Chatter
No

Signature
public static ConnectApi.AudienceCollection getAudiences(String communityId, String ipAddress, String domain, String userId, ConnectApi.PublishStatus publishStatus, Boolean includeAudienceCriteria, List<String> targetTypes, String recordId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

ipAddress
Type: String
IP address of the user. If null, no audiences with location criteria are returned.

domain
Type: String
Name of the user's Salesforce custom domain. If null, no audiences with domain criteria are returned.

userId
Type: String
ID of the user. If null, defaults to the ID of the context user.

publishStatus
Type: ConnectApi.PublishStatus
Publish status of the audience. Values are:
• Draft
• Live
If null, defaults to Live.

includeAudienceCriteria
Type: Boolean
Specifies whether to include audience criteria (true) or not (false). If unspecified, defaults to false.

targetTypes
Type: List<String>
Comma-separated list of target types to filter the results. Supported values include:
• ExperienceVariation (version 48.0 and later)
- Custom object API names, such as `CustomObjectName__c` (version 48.0 and later)
- NavigationLinkSet (version 49.0 and later)
- Topic (version 49.0 and later)
- CollaborationGroup (version 49.0 and later)
- KnowledgeArticle (version 49.0 and later)
- ContentDocument (version 49.0 and later)
- ManagedContent (version 49.0 and later)
- Report (version 49.0 and later)
- Dashboard (version 49.0 and later)

If `null`, all target types are returned.

`recordId`
- Type: `String`
  - ID of the record for field based criteria. If `null`, all applicable audiences with field based criteria are returned.

**Return Value**
Type: `ConnectApi.AudienceCollection`

```java
getAudiences(communityId, ipAddress, domain, userId, publishStatus, includeAudienceCriteria, targetTypes)
```

Get a list of assigned audiences that match the user context.

**API Version**
48.0

**Available to Guest Users**
48.0

**Requires Chatter**
No

**Signature**
```java
public static ConnectApi.AudienceCollection getAudiences(String communityId, String ipAddress, String domain, String userId, ConnectApi.PublishStatus publishStatus, Boolean includeAudienceCriteria, List<String> targetTypes)
```

**Parameters**

- `communityId`
  - Type: `String`
  - ID of the Experience Cloud site.
**ipAddress**
- **Type:** String
  - IP address of the user. If null, no audiences with location criteria are returned.

**domain**
- **Type:** String
  - Name of the user's Salesforce custom domain. If null, no audiences with domain criteria are returned.

**userId**
- **Type:** String
  - ID of the user. If null, defaults to the ID of the context user.

**publishStatus**
- **Type:** ConnectApi.PublishStatus
  - Publish status of the audience. Values are:
    - Draft
    - Live
  - If null, defaults to Live.

**includeAudienceCriteria**
- **Type:** Boolean
  - Specifies whether to include audience criteria (true) or not (false). If unspecified, defaults to false.

**targetTypes**
- **Type:** List<String>
  - Comma-separated list of target types to filter the results. Supported values include:
    - ExperienceVariation (version 48.0 and later)
    - Custom object API names, such as CustomObjectAName__c (version 48.0 and later)
    - NavigationLinkSet (version 49.0 and later)
    - Topic (version 49.0 and later)
    - CollaborationGroup (version 49.0 and later)
    - KnowledgeArticle (version 49.0 and later)
    - ContentDocument (version 49.0 and later)
    - ManagedContent (version 49.0 and later)
    - Report (version 49.0 and later)
    - Dashboard (version 49.0 and later)
  - If null, all target types are returned.

**Return Value**
- **Type:** ConnectApi.AudienceCollection

**getTarget(communityId, targetId)**
- Get a target.
API Version
48.0

Available to Guest Users
48.0

Requires Chatter
No

Signature
public static ConnectApi.Target getTarget(String communityId, String targetId)

Parameters

communityId
Type: String
ID of the Experience Cloud site.

targetId
Type: String
ID of the target.

Return Value
Type: ConnectApi.Target

getTargetBatch(communityId, targetIds)
Get target information for a comma-separated list of target IDs.

API Version
48.0

Available to Guest Users
48.0

Requires Chatter
No

Signature
public static ConnectApi.BatchResult[] getTargetBatch(String communityId, List<String> targetIds)
Parameters

communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

targetIds
  Type: List<String>
  Comma-separated list of target IDs.

Return Value

Type: ConnectApi.BatchResult
The ConnectApi.BatchResult.getResult() method returns a ConnectApi.Target object and errors for targets that didn't load.

getTargets(communityId, ipAddress, domain, userId, publishStatus, recordId, targetTypes, includeAudience, includeAllMatchingTargetsWithinGroup, groupNames)
Get a list of targets that match the user context, based on the assigned audiences that include the user.

API Version

48.0

Available to Guest Users

48.0

Requires Chatter

No

Signature

public static ConnectApi.TargetCollection getTargets(String communityId, String ipAddress, String domain, String userId, ConnectApi.PublishStatus publishStatus, String recordId, List<String> targetTypes, Boolean includeAudience, Boolean includeAllMatchingTargetsWithinGroup, List<String> groupNames)

Parameters

communityId
  Type: String
  ID of the Experience Cloud site.

ipAddress
  Type: String
  IP address of the user. If null, no audiences with location criteria are returned.
domain
Type: String
Name of the user's Salesforce custom domain. If null, no audiences with domain criteria are returned.

userId
Type: String
ID of the user. If null, the default is the ID of the context user.

publishStatus
Type: ConnectApi.PublishStatus
Publish status of the target. Values are:
- Draft
- Live

recordId
Type: String
ID of the record, if you want to specify field based criteria in audiences.

targetTypes
Type: List<String>
Comma-separated list of target types to filter the results. Supported values include:
- ExperienceVariation (version 48.0 and later)
- Custom object API names, such as CustomObjectName__c (version 48.0 and later)
- NavigationLinkSet (version 49.0 and later)
- Topic (version 49.0 and later)
- CollaborationGroup (version 49.0 and later)
- KnowledgeArticle (version 49.0 and later)
- ContentDocument (version 49.0 and later)
- ManagedContent (version 49.0 and later)
- Report (version 49.0 and later)
- Dashboard (version 49.0 and later)
If null, all target types are returned.

includeAudience
Type: Boolean
Specifies whether to include the matching audience (true) or not (false). If null, the default is false.

includeAllMatchingTargetsWithinGroup
Type: Boolean
Specifies whether to include all the matching targets within a target group (true) or not (false). If null, the default is false.
If false, the first matching target within each group, based on priority within the group, is returned.

groupNames
Type: List<String>
A comma-separated list of group names. Groups bundle related target and audience pairs.
Return Value
Type: ConnectApi.TargetCollection

updateAudience(communityId, audienceId, audience)
Update an audience.

API Version
48.0

Requires Chatter
No

Signature
public static ConnectApi.Audience updateAudience(String communityId, String audienceId, ConnectApi.AudienceInput audience)

Parameters
communityId
  Type: String
  ID of the Experience Cloud site.

audienceId
  Type: String
  ID of the audience.

audience
  Type: ConnectApi.AudienceInput
  A ConnectApi.AudienceInput object that defines the updates to the audience.

Return Value
Type: ConnectApi.Audience

updateTargets(communityId, target)
Update targets.

API Version
48.0

Requires Chatter
No
Signature

public static ConnectApi.TargetCollection updateTargets(String communityId, ConnectApi.TargetCollectionUpdateInput target)

Parameters

communityId
Type: String
ID of the Experience Cloud site.

target
Type: ConnectApi.TargetCollectionUpdateInput
A ConnectApi.TargetCollectionUpdateInput object that defines the updates for the targets.

Return Value
Type: ConnectApi.TargetCollection

PickTicket Class

Create tickets to fulfill orders.

Namespace

ConnectApi

Pick Ticket Methods

These methods are for Pick Tickets. All methods are static.

IN THIS SECTION:

distributePickedQuantities(distributePickedQuantitiesInput)
Distribute picked quantities among orders in a pick ticket.

distributePickedQuantities (distributePickedQuantitiesInput)
Distribute picked quantities among orders in a pick ticket.

API Version
58.0

Requires Chatter
No
Signature

```java
public static ConnectApi.DistributePickedQuantitiesOutputRepresentation
distributePickedQuantities(ConnectApi.DistributePickedQuantitiesInputRepresentation
distributePickedQuantitiesInput)
```

Parameters

distributePickedQuantitiesInput
  Type: ConnectApi.DistributePickedQuantitiesInputRepresentation
  Input to distribute picked quantities.

Return Value

Type: ConnectApi.DistributePickedQuantitiesOutputRepresentation

**QuestionAndAnswers Class**

*Access question and answers suggestions.*

**Namespace**

*ConnectApi*

**IN THIS SECTION:**

**QuestionAndAnswers Methods**

The following are methods for **QuestionAndAnswers**. All methods are static.

**QuestionAndAnswers Methods**

The following are methods for **QuestionAndAnswers**. All methods are static.

**IN THIS SECTION:**

- `getSuggestions(communityId, q, subjectId, includeArticles, maxResults)`
  Get question and answers suggestions.

- `setTestGetSuggestions(communityId, q, subjectId, includeArticles, maxResults, result)`
  Register a `ConnectApi.QuestionAndAnswersSuggestions` object to be returned when `getSuggestions` is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

- `updateQuestionAndAnswers(communityId, feedElementId, questionAndAnswersCapability)`
  Choose or change the best answer for a question.

**getSuggestions(communityId, q, subjectId, includeArticles, maxResults)**

Get question and answers suggestions.
API Version
32.0

Requires Chatter
Yes

Signature
public static ConnectApi.QuestionAndAnswersSuggestions getSuggestions(String communityId, String q, String subjectId, Boolean includeArticles, Integer maxResults)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
Required and can’t be null. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

subjectId
Type: String
Specify a subject ID to search only questions on that object. If the ID is a topic or a user, the ID is ignored.

includeArticles
Type: Boolean
Specify true to include knowledge articles in the search results. To return only questions, specify false.

maxResults
Type: Integer
The maximum number of results to return for each type of item. Possible values are 1–10. The default value is 5.

Return Value
Type: ConnectApi.QuestionAndAnswersSuggestions

Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetSuggestions(communityId, q, subjectId, includeArticles, maxResults, result)
**setTestGetSuggestions(communityId, q, subjectId, includeArticles, maxResults, result)**

Register a `ConnectApi.QuestionAndAnswersSuggestions` object to be returned when `getSuggestions` is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

**API Version**

32.0

**Signature**

```java
public static Void setTestGetSuggestions(String communityId, String q, String subjectId, Boolean includeArticles, Integer maxResults, ConnectApi.QuestionAndAnswersSuggestions result)
```

**Parameters**

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **q**
  - Type: `String`
  - Required and can't be `null`. Specifies the string to search. The search string must contain at least two characters, not including wildcards. See [Wildcards](#).

- **subjectId**
  - Type: `String`
  - Specify a subject ID to search only questions on that object. If the ID is a topic or a user, the ID is ignored.

- **includeArticles**
  - Type: `Boolean`
  - Specify `true` to include knowledge articles in the search results. To return only questions, specify `false`.

- **maxResults**
  - Type: `Integer`
  - The maximum number of results to return for each type of item. Possible values are 1–10. The default value is 5.

- **result**
  - Type: `ConnectApi.QuestionAndAnswersSuggestions`
  - Object containing test data.

**Return Value**

Type: `Void`  

**SEE ALSO:**

- `getSuggestions(communityId, q, subjectId, includeArticles, maxResults)`
- [Apex Developer Guide: Testing ConnectApi Code](#)
updateQuestionAndAnswers(communityId, feedElementId, questionAndAnswersCapability)

Choose or change the best answer for a question.

API Version
32.0

Requires Chatter
Yes

Signature
public static ConnectApi.QuestionAndAnswersCapability updateQuestionAndAnswers(String communityId, String feedElementId, ConnectApi.QuestionAndAnswersCapabilityInput questionAndAnswersCapability)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

feedElementId
Type: String
ID of the feed element.

questionAndAnswersCapability
Type: ConnectApi.QuestionAndAnswersCapabilityInput
Specify the best answer (comment ID) for the question.

Return Value
Type: ConnectApi.QuestionAndAnswersCapability
If the feed element doesn’t support this capability, the return value is ConnectApi.NotFoundException.

Example

ConnectApi.QuestionAndAnswersCapabilityInput qaInput = new ConnectApi.QuestionAndAnswersCapabilityInput();
qaInput.bestAnswerId = '0D7D00000000lMAKAY';

ConnectApi.QuestionAndAnswersCapability qa =
ConnectApi.QuestionAndAnswers.updateQuestionAndAnswers(null, '0D5D0000000XZjJ', qaInput);
Recommendations Class

Get and reject Chatter, custom, and static recommendations. Create, get, update, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

For Next Best Action recommendations, see NextBestAction Class.

Namespace

ConnectApi

Recommendations Methods

The following are methods for Recommendations. All methods are static.

IN THIS SECTION:

- createRecommendationAudience(communityId, recommendationAudience)
  Create an audience for a custom recommendation.
- createRecommendationAudience(communityId, name)
  Create an audience for a custom recommendation.
- createRecommendationDefinition(communityId, recommendationDefinition)
  Create a custom recommendation definition.
- createRecommendationDefinition(communityId, name, title, actionUrl, actionUrlName, explanation)
  Create a custom recommendation definition with the specified parameters.
- createScheduledRecommendation(communityId, scheduledRecommendation)
  Create a scheduled custom recommendation.
- createScheduledRecommendation(communityId, recommendationDefinitionId, rank, enabled, recommendationAudienceId, channel)
  Create a scheduled custom recommendation with the specified parameters.
- deleteRecommendationAudience(communityId, recommendationAudienceId)
  Delete a custom recommendation audience.
- deleteRecommendationDefinition(communityId, recommendationDefinitionId)
  Delete a custom recommendation definition.
- deleteRecommendationDefinitionPhoto(communityId, recommendationDefinitionId)
  Delete a custom recommendation definition photo.
- deleteScheduledRecommendation(communityId, scheduledRecommendationId, deleteDefinitionIfLast)
  Delete a scheduled custom recommendation.
- getRecommendationAudience(communityId, recommendationAudienceId)
  Get information about a custom recommendation audience.
- getRecommendationAudienceMembership(communityId, recommendationAudienceId)
  Get the members of a custom recommendation audience.
- getRecommendationAudienceMembership(communityId, recommendationAudienceId, pageParam, pageSize)
  Get a page of custom recommendation audience members.
getRecommendationAudiences(communityId)
Get custom recommendation audiences.

getRecommendationAudiences(communityId, pageParam, pageSize)
Get a page of custom recommendation audiences.

getRecommendationDefinition(communityId, recommendationDefinitionId)
Get a custom recommendation definition.

getRecommendationDefinitionPhoto(communityId, recommendationDefinitionId)
Get a custom recommendation definition photo.

getRecommendationDefinitions(communityId)
Get custom recommendation definitions.

getRecommendationForUser(communityId, userId, action, objectId)
Get the Chatter, custom, or static recommendation for the context user for the specified action and object ID.

getRecommendationsForUser(communityId, userId, contextAction, contextObjectId, channel, maxResults)
Get the Chatter recommendations, such as user, group, file, article, record, and topic recommendations for the context user. Get the custom and static recommendations for the context user.

getRecommendationsForUser(communityId, userId, action, contextAction, contextObjectId, channel, maxResults)
Get the Chatter, custom, and static recommendations for the context user for the specified action.

getRecommendationsForUser(communityId, userId, action, objectCategory, contextAction, contextObjectId, channel, maxResults)
Get the Chatter, custom, and static recommendations for the context user for the specified action and object category.

getScheduledRecommendation(communityId, scheduledRecommendationId)
Get a scheduled custom recommendation.

getScheduledRecommendations(communityId, channel)
Get scheduled custom recommendations.

rejectRecommendationForUser(communityId, userId, action, objectId)
Reject a Chatter, custom, or static recommendation for the context user for the specified action and object ID.

rejectRecommendationForUser(communityId, userId, action, objectEnum)
Reject a static recommendation for the context user.

updateRecommendationAudience(communityId, recommendationAudienceId, recommendationAudience)
Update a custom recommendation audience.

updateRecommendationDefinition(communityId, recommendationDefinitionId, name, title, actionUrl, actionUrlName, explanation)
Update a custom recommendation definition with the specified parameters.

updateRecommendationDefinition(communityId, recommendationDefinitionId, recommendationDefinition)
Update a custom recommendation definition.

updateRecommendationDefinitionPhoto(communityId, recommendationDefinitionId, fileUpload)
Update a custom recommendation definition photo with a file that hasn’t been uploaded.

updateRecommendationDefinitionPhoto(communityId, recommendationDefinitionId, fileId, versionNumber)
Update a custom recommendation definition photo with an uploaded file.

updateRecommendationDefinitionPhotoWithAttributes(communityId, recommendationDefinitionId, photo)
Update a custom recommendation definition photo with an uploaded file that requires cropping.

updateRecommendationDefinitionPhotoWithAttributes(communityId, recommendationDefinitionId, photo, fileUpload)
Update a custom recommendation definition photo with a file that hasn’t been uploaded and requires cropping.
**updateScheduledRecommendation(communityId, scheduledRecommendationId, scheduledRecommendation)**

Update a scheduled custom recommendation.

**updateScheduledRecommendation(communityId, scheduledRecommendationId, rank, enabled, recommendationAudienceId)**

Update a scheduled custom recommendation with the specified parameters.

---

**createRecommendationAudience(communityId, recommendationAudience)**

Create an audience for a custom recommendation.

**API Version**

35.0

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.RecommendationAudience createRecommendationAudience(String communityId, ConnectApi.RecommendationAudienceInput recommendationAudience)
```

**Parameters**

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or null.

- **recommendationAudience**
  - Type: `ConnectApi.RecommendationAudienceInput`

**Return Value**

Type: `ConnectApi.RecommendationAudience`

**Usage**

Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

**createRecommendationAudience(communityId, name)**

Create an audience for a custom recommendation.

**API Version**

35.0
Requires Chatter
Yes

Signature

```java
public static ConnectApi.RecommendationAudience createRecommendationAudience(String communityId, String name)
```  

Parameters

- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.

- `name`
  Type: `String`
  Name of the audience.

Return Value

Type: `ConnectApi.RecommendationAudience`

Usage

Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

```java
createRecommendationDefinition(communityId, recommendationDefinition)
```

Create a custom recommendation definition.

API Version

35.0

Requires Chatter
Yes

Signature

```java
public static ConnectApi.RecommendationDefinition createRecommendationDefinition(String communityId, ConnectApi.RecommendationDefinitionInput recommendationDefinition)
```  

Parameters

- `communityId`
  Type: `String`
ID for an Experience Cloud site, internal, or null.

**recommendationDefinition**
Type: `ConnectApi.RecommendationDefinitionInput`  
A `ConnectApi.RecommendationDefinitionInput` object.

**Return Value**
Type: `ConnectApi.RecommendationDefinition`

**Usage**
Recommendation definitions allow you to create custom recommendations that appear in Experience Cloud sites, encouraging users to watch videos, take training and more.

Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

These recommendations appear by default on the Customer Service template. They appear on the home and question detail pages and in the feed in Salesforce mobile web. They also appear anywhere community managers add recommendations using Experience Builder in the Customer Service template.

So that users don’t see the same recommendations all the time, Salesforce periodically removes and brings back custom recommendations that haven’t been accepted or dismissed.

```java
createRecommendationDefinition(communityId, name, title, actionUrl, actionUrlName, explanation)
```
Create a custom recommendation definition with the specified parameters.

**API Version**
35.0

**Requires Chatter**
Yes

**Signature**
```java
public static ConnectApi.RecommendationDefinition createRecommendationDefinition(String communityId, String name, String title, String actionUrl, String actionUrlName, String explanation)
```

**Parameters**
- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.  

name
Type: String
Name of the custom recommendation definition. The name is displayed in Setup.

title
Type: String
Title of the custom recommendation definition.

actionUrl
Type: String
URL for acting on the custom recommendation, for example, the URL to join a group.

actionUrlName
Type: String
Text label for the action URL in the user interface, for example, “Launch.”

explanation
Type: String
Explanation, or body, of the custom recommendation.

Return Value
Type: ConnectApi.RecommendationDefinition

Usage
Recommendation definitions allow you to create custom recommendations that appear in Experience Cloud sites, encouraging users to watch videos, take training and more.

Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

These recommendations appear by default on the Customer Service template. They appear on the home and question detail pages and in the feed in Salesforce mobile web. They also appear anywhere community managers add recommendations using Experience Builder in the Customer Service template.

So that users don’t see the same recommendations all the time, Salesforce periodically removes and brings back custom recommendations that haven’t been accepted or dismissed.

createScheduledRecommendation(communityId, scheduledRecommendation)
Create a scheduled custom recommendation.

API Version
35.0

Requires Chatter
Yes
Signature

```java
public static ConnectApi.ScheduledRecommendation createScheduledRecommendation(String communityId, ConnectApi.ScheduledRecommendationInput scheduledRecommendation)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or null.

- `scheduledRecommendation`
  - Type: `ConnectApi.ScheduledRecommendationInput`

Return Value

- Type: `ConnectApi.ScheduledRecommendation`

Usage

Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

```java
createScheduledRecommendation(communityId, recommendationDefinitionId, rank, enabled, recommendationAudienceId, channel)
```
Create a scheduled custom recommendation with the specified parameters.

API Version

36.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.ScheduledRecommendation createScheduledRecommendation(String communityId, String recommendationDefinitionId, Integer rank, Boolean enabled, String recommendationAudienceId, ConnectApi.RecommendationChannel channel)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or null.
**recommendationDefinitionId**

Type: **String**

ID of the custom recommendation definition.

**rank**

Type: **Integer**

Relative rank of the scheduled custom recommendation indicated by ascending whole numbers starting with 1.

Setting the rank is comparable to an insertion into an ordered list. The scheduled custom recommendation is inserted into the position specified by the `rank`. The `rank` of all the scheduled custom recommendations after it is pushed down. See Ranking scheduled custom recommendations example.

If the specified `rank` is larger than the size of the list, the scheduled custom recommendation is put at the end of the list. The `rank` of the scheduled custom recommendation is the size of the list, instead of the one specified.

If a `rank` is not specified, the scheduled custom recommendation is put at the end of the list.

**enabled**

Type: **Boolean**

Indicates whether scheduling is enabled. If `true`, the custom recommendation is enabled and appears in Experience Cloud sites. If `false`, custom recommendations in feeds in Salesforce mobile web aren’t removed, but no new custom recommendations appear. In Customer Service and Partner Central sites, disabled custom recommendations no longer appear.

**recommendationAudienceId**

Type: **String**

ID of the custom recommendation definition that this scheduled recommendation schedules.

**channel**

Type: **ConnectApi.RecommendationChannel**

A way to tie custom recommendations together. For example, display recommendations in specific places in the UI or show recommendations based on time of day or geographic locations. Values are:

- **CustomChannel1**—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels. For example, community managers can use Experience Builder to determine where recommendations appear.
- **CustomChannel2**—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
- **CustomChannel3**—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
- **CustomChannel4**—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
- **CustomChannel5**—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
- **DefaultChannel**—Default recommendation channel. Recommendations appear by default on the Home and Question Detail pages of Customer Service and Partner Central Experience Builder templates. They also appear in the feed in the Salesforce mobile web and anywhere community managers add recommendations using Experience Builder.

Use these channel values; you can’t rename or create other channels.

**Return Value**

Type: **ConnectApi.ScheduledRecommendation**
**Usage**

Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

**Ranking scheduled custom recommendations example**

If you have these scheduled custom recommendations:

<table>
<thead>
<tr>
<th>Scheduled Recommendations</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>ScheduledRecommendationA</td>
<td>1</td>
</tr>
<tr>
<td>ScheduledRecommendationB</td>
<td>2</td>
</tr>
<tr>
<td>ScheduledRecommendationC</td>
<td>3</td>
</tr>
</tbody>
</table>

And you include this information in the Scheduled Custom Recommendation Input:

<table>
<thead>
<tr>
<th>Scheduled Recommendation</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>ScheduledRecommendationD</td>
<td>2</td>
</tr>
</tbody>
</table>

The result is:

<table>
<thead>
<tr>
<th>Scheduled Recommendation</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>ScheduledRecommendationA</td>
<td>1</td>
</tr>
<tr>
<td>ScheduledRecommendationD</td>
<td>2</td>
</tr>
<tr>
<td>ScheduledRecommendationB</td>
<td>3</td>
</tr>
<tr>
<td>ScheduledRecommendationC</td>
<td>4</td>
</tr>
</tbody>
</table>

**deleteRecommendationAudience(communityId, recommendationAudienceId)**

Delete a custom recommendation audience.

**API Version**

35.0

**Requires Chatter**

Yes
Signature

```java
public static Void deleteRecommendationAudience(String communityId, String recommendationAudienceId)
```

Parameters

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **recommendationAudienceId**
  - Type: `String`
  - ID of the custom recommendation audience.

Return Value

Type: `Void`

Usage

Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

**deleteRecommendationDefinition(String communityId, String recommendationDefinitionId)**

Delete a custom recommendation definition.

API Version

35.0

Requires Chatter

Yes

Signature

```java
public static Void deleteRecommendationDefinition(String communityId, String recommendationDefinitionId)
```

Parameters

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **recommendationDefinitionId**
  - Type: `String`
ID of the custom recommendation definition.

Return Value
Type: Void

Usage
Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

deleteRecommendationDefinitionPhoto(communityId, recommendationDefinitionId)
Delete a custom recommendation definition photo.

API Version
35.0

Requires Chatter
Yes

Signature
public static Void deleteRecommendationDefinitionPhoto(String communityId, String recommendationDefinitionId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

recommendationDefinitionId
Type: String
ID of the custom recommendation definition.

Return Value
Type: Void

Usage
Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.
deleteScheduledRecommendation(communityId, scheduledRecommendationId, deleteDefinitionIfLast)

Delete a scheduled custom recommendation.

API Version
35.0

Requires Chatter
Yes

Signature
public static Void deleteScheduledRecommendation(String communityId, String scheduledRecommendationId, Boolean deleteDefinitionIfLast)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

scheduledRecommendationId
Type: String
ID of the scheduled custom recommendation.

deleteDefinitionIfLast
Type: Boolean
If true and if this is the last scheduled custom recommendation of a custom recommendation definition, deletes the custom recommendation definition. Default is false.

Return Value
Type: Void

Usage
Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

Deleting a scheduled custom recommendation is comparable to a deletion in an ordered list. All scheduled custom recommendations after the deleted scheduled custom recommendation receive a new, higher rank automatically.

getRecommendationAudience(communityId, recommendationAudienceId)

Get information about a custom recommendation audience.
API Version
35.0

Requires Chatter
Yes

Signature
public static ConnectApi.RecommendationAudience getRecommendationAudience(String communityId, String recommendationAudienceId)

Parameters
communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.
recommendationAudienceId
  Type: String
  ID of the custom recommendation audience.

Return Value
Type: ConnectApi.RecommendationAudience

Usage
Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

getRecommendationAudienceMembership(communityId, recommendationAudienceId)
Get the members of a custom recommendation audience.

API Version
35.0

Requires Chatter
Yes

Signature
public static ConnectApi.UserReferencePage getRecommendationAudienceMembership(String communityId, String recommendationAudienceId)
Parameters

`communityId`

Type: `String`

ID for an Experience Cloud site, internal, or null.

`recommendationAudienceId`

Type: `String`

ID of the custom recommendation audience.

Return Value

Type: `ConnectApi.UserReferencePage`

Usage

Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

`getRecommendationAudienceMembership(communityId, recommendationAudienceId, pageParam, pageSize)`

Get a page of custom recommendation audience members.

API Version

35.0

Requires Chatter

Yes

Signature

`public static ConnectApi.UserReferencePage getRecommendationAudienceMembership(String communityId, String recommendationAudienceId, Integer pageParam, Integer pageSize)`

Parameters

`communityId`

Type: `String`

ID for an Experience Cloud site, internal, or null.

`recommendationAudienceId`

Type: `String`

ID of the custom recommendation audience.

`pageParam`

Type: `Integer`
Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

`pageSize`
Type: `Integer`
Specifies the number of members per page.

Return Value
Type: `ConnectApi.UserReferencePage`

Usage
Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

`getRecommendationAudiences(communityId)`
Get custom recommendation audiences.

API Version
35.0

Requires Chatter
Yes

Signature
```java
public static ConnectApi.RecommendationAudiencePage getRecommendationAudiences(String communityId)
```

Parameters
`communityId`
Type: `String`
ID for an Experience Cloud site, `internal`, or `null`.

Return Value
Type: `ConnectApi.RecommendationAudiencePage`

Usage
Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.
getRecommendationAudiences(communityId, pageParam, pageSize)

Get a page of custom recommendation audiences.

API Version
35.0

Requires Chatter
Yes

Signature
public static ConnectApi.RecommendationAudiencePage getRecommendationAudiences(String communityId, Integer pageParam, Integer pageSize)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
Type: Integer
Specifies the number of audiences per page.

Return Value
Type: ConnectApi.RecommendationAudiencePage

Usage
Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

getRecommendationDefinition(communityId, recommendationDefinitionId)

Get a custom recommendation definition.

API Version
35.0
Requires Chatter
Yes

Signature

```java
public static ConnectApi.RecommendationDefinition getRecommendationDefinition(String communityId, String recommendationDefinitionId)
```

Parameters

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.
- **recommendationDefinitionId**
  - Type: String
  - ID of the custom recommendation definition.

Return Value

Type: ConnectApi.RecommendationDefinition

Usage

Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

getRecommendationDefinitionPhoto(communityId, recommendationDefinitionId)

Get a custom recommendation definition photo.

API Version

35.0

Requires Chatter
Yes

Signature

```java
public static ConnectApi.Photo getRecommendationDefinitionPhoto(String communityId, String recommendationDefinitionId)
```

Parameters

- **communityId**
  - Type: String
ID for an Experience Cloud site, internal, or null.

`recommendationDefinitionId`

Type: `String`

ID of the custom recommendation definition.

**Return Value**

Type: `ConnectApi.RecommendationDefinitionPage`

**Usage**

Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

### `getRecommendationDefinitions(communityId)`

Get custom recommendation definitions.

**API Version**

35.0

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.RecommendationDefinitionPage getRecommendationDefinitions(String communityId)
```

**Parameters**

`communityId`

Type: `String`

ID for an Experience Cloud site, internal, or null.

**Return Value**

Type: `ConnectApi.RecommendationDefinitionPage`

**Usage**

Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.
getRecommendationForUser(communityId, userId, action, objectId)
Get the Chatter, custom, or static recommendation for the context user for the specified action and object ID.

API Version
33.0

Requires Chatter
Yes

Signature
public static ConnectApi.RecommendationCollection getRecommendationForUser(String communityId, String userId, ConnectApi.RecommendationActionType action, String objectId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

userId
Type: String
ID for the context user or the keyword me.

action
Type: ConnectApi.RecommendationActionType
Specifies the action to take on a recommendation.
- follow—Follow a file, record, topic, or user.
- join—Join a group.
- view—View a file, group, article, record, user, custom, or static recommendation.

objectId
Type: String
Specifies the object to act on.
- If action is follow, objectId is a user ID, file ID, record ID, or topic ID (version 36.0 and later).
- If action is join, objectId is a group ID.
- If action is view, objectId is a user ID, file ID, group ID, record ID, custom recommendation ID (version 34.0 and later), the enum Today for static recommendations (version 35.0 and later), or an article ID (version 37.0 and later).

Return Value
Type: ConnectApi.RecommendationCollection
**Usage**

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- `setTestGetRecommendationForUser(communityId, userId, action, objectId, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

---

**getRecommendationsForUser**

```java
public static ConnectApi.RecommendationCollection getRecommendationsForUser(String communityId, String userId, ConnectApi.RecommendationActionType contextAction, String contextObjectId, ConnectApi.RecommendationChannel channel, Integer maxResults)
```

Get the Chatter recommendations, such as user, group, file, article, record, and topic recommendations for the context user. Get the custom and static recommendations for the context user.

---

**API Version**

36.0

---

**Available to Guest Users**

38.0

⚠️ **Note:** Only article and file recommendations are available to guest users.

---

**Requires Chatter**

Yes

---

**Signature**

```java
public static ConnectApi.RecommendationCollection getRecommendationsForUser(String communityId, String userId, ConnectApi.RecommendationActionType contextAction, String contextObjectId, ConnectApi.RecommendationChannel channel, Integer maxResults)
```

**Parameters**

- **communityId**
  - **Type:** `String`
  - ID for an Experience Cloud site, internal, or null.

- **userId**
  - **Type:** `String`
  - ID for the context user or the keyword `me`.

- **contextAction**
  - **Type:** `ConnectApi.RecommendationActionType`
  - Action that the context user just performed. Supported values are:
    - `follow`
• view

Use contextAction and contextObjectId together to get new recommendations based on the action just performed. If you don’t want recommendations based on a recent action, specify null.

contextObjectId
Type: String
ID of the object that the context user just performed an action on.
• If contextAction is follow, contextObjectId is a user ID, file ID, record ID, or topic ID.
• If contextAction is view, contextObjectId is a user ID, file ID, group ID, record ID, or article ID (version 37.0 and later).

Use contextAction and contextObjectId together to get new recommendations based on the action just performed. If you don’t want recommendations based on a recent action, specify null.

channel
Type: ConnectApi.RecommendationChannel
A way to tie custom recommendations together. For example, display recommendations in specific places in the UI or show recommendations based on time of day or geographic locations. Values are:
• CustomChannel1—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels. For example, community managers can use Experience Builder to determine where recommendations appear.
• CustomChannel2—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
• CustomChannel3—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
• CustomChannel4—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
• CustomChannel5—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
• DefaultChannel—Default recommendation channel. Recommendations appear by default on the Home and Question Detail pages of Customer Service and Partner Central Experience Builder templates. They also appear in the feed in the Salesforce mobile web and anywhere community managers add recommendations using Experience Builder.

maxResults
Type: Integer
Maximum number of recommendation results; default is 10. Values must be from 1 to 99.

Return Value
Type: ConnectApi.RecommendationCollection

Usage
If you want to get recommendations based on a recent action performed, such as following a user, use contextAction and contextObjectId together. For example, if you just followed Pam, you specify follow for contextAction and Pam’s user ID for contextObjectId.

This method only recommends users who are followed by people who follow Pam. In this example, John follows Pam so the method returns a recommendation for you to follow Suzanne since John also follows Suzanne.
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- setTestGetRecommendationsForUser(communityId, userId, contextAction, contextObjectId, channel, maxResults, result)

getRecommendationsForUser(communityId, userId, action, contextAction, contextObjectId, channel, maxResults)
Get the Chatter, custom, and static recommendations for the context user for the specified action.

API Version
36.0

Available to Guest Users
38.0

Note: Only article and file recommendations are available to guest users.

Requires Chatter
Yes

Signature
public static ConnectApi.RecommendationCollection getRecommendationsForUser(String communityId, String userId, ConnectApi.RecommendationActionType action, ConnectApi.RecommendationActionType contextAction, String contextObjectId, ConnectApi.RecommendationChannel channel, Integer maxResults)

Parameters
  communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.
  userId
  Type: String
  ID for the context user or the keyword me.
  action
  Type: ConnectApi.RecommendationActionType
  Specifies the action to take on a recommendation.
  • follow—Follow a file, record, topic, or user.
  • join—Join a group.
  • view—View a file, group, article, record, user, custom, or static recommendation.
**contextAction**

Type: `ConnectApi.RecommendationActionType`

Action that the context user just performed. Supported values are:

- follow
- view

Use `contextAction` and `contextObjectId` together to get new recommendations based on the action just performed. If you don’t want recommendations based on a recent action, specify `null`.

**contextObjectId**

Type: `String`

ID of the object that the context user just performed an action on.

- If `contextAction` is `follow`, `contextObjectId` is a user ID, file ID, record ID, or topic ID.
- If `contextAction` is `view`, `contextObjectId` is a user ID, file ID, group ID, record ID, or article ID (version 37.0 and later).

Use `contextAction` and `contextObjectId` together to get new recommendations based on the action just performed. If you don’t want recommendations based on a recent action, specify `null`.

**channel**

Type: `ConnectApi.RecommendationChannel`

A way to tie custom recommendations together. For example, display recommendations in specific places in the UI or show recommendations based on time of day or geographic locations. Values are:

- CustomChannel1—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels. For example, community managers can use Experience Builder to determine where recommendations appear.
- CustomChannel2—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
- CustomChannel3—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
- CustomChannel4—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
- CustomChannel5—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.

**maxResults**

Type: `Integer`

Maximum number of recommendation results; default is 10. Values must be from 1 to 99.

**Return Value**

Type: `ConnectApi.RecommendationCollection`
Usage
If you want to get recommendations based on a recent action performed, such as following a user, use `contextAction` and `contextObjectId` together. For example, if you just followed Pam, you specify `follow` for `contextAction` and Pam’s user ID for `contextObjectId`.

This method only recommends users who are followed by people who follow Pam. In this example, John follows Pam so the method returns a recommendation for you to follow Suzanne since John also follows Suzanne.

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

```java
setTestGetRecommendationsForUser(communityId, userId, action, contextAction, contextObjectId, channel, maxResults, result)
```

*Apex Developer Guide: Testing ConnectApi Code*

### getRecommendationsForUser(communityId, userId, action, objectCategory, contextAction, contextObjectId, channel, maxResults)

Get the Chatter, custom, and static recommendations for the context user for the specified action and object category.

**API Version**

36.0

**Available to Guest Users**

38.0

⚠️ **Note:** Only article and file recommendations are available to guest users.

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.RecommendationCollection getRecommendationsForUser(String communityId, String userId, ConnectApi.RecommendationActionType action, String objectCategory, ConnectApi.RecommendationActionType contextAction, String contextObjectId, ConnectApi.RecommendationChannel channel, Integer maxResults)
```

**Parameters**

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, `internal`, or `null`.

- `userId`
  - Type: `String`
  - ID for the context user or the keyword `me`.
action
Type: ConnectApi.RecommendationActionType
Specifies the action to take on a recommendation.
• follow—Follow a file, record, topic, or user.
• join—Join a group.
• view—View a file, group, article, record, user, custom, or static recommendation.

objectCategory
Type: String
• If action is follow, objectCategory is users, files, topics, or records.
• If action is join, objectCategory is groups.
• If action is view, objectCategory is users, files, groups, records, custom, apps, or articles (version 37.0 and later).

You can also specify a key prefix, the first three characters of the object ID, as the objectCategory. Valid values are:
• If action is follow, objectCategory is 005 (users), 069 (files), 0TO (topics), or 001 (accounts), for example.
• If action is join, objectCategory is 0F9 (groups).
• If action is view, objectCategory is 005 (users), 069 (files), 0F9 (groups), 0RD (custom recommendations), T (static recommendations), 001 (accounts), or kA0 (articles), for example, (version 37.0 and later).

collectionAction
Type: ConnectApi.RecommendationActionType
Action that the context user just performed. Supported values are:
• follow
• view

Use contextAction and contextObjectId together to get new recommendations based on the action just performed. If you don't want recommendations based on a recent action, specify null.

collectionObjectId
Type: String
ID of the object that the context user just performed an action on.
• If contextAction is follow, collectionObjectId is a user ID, file ID, record ID, or topic ID.
• If contextAction is view, collectionObjectId is a user ID, file ID, group ID, record ID, or article ID (version 37.0 and later).

Use contextAction and contextObjectId together to get new recommendations based on the action just performed. If you don't want recommendations based on a recent action, specify null.

channel
Type: ConnectApi.RecommendationChannel
A way to tie custom recommendations together. For example, display recommendations in specific places in the UI or show recommendations based on time of day or geographic locations. Values are:
• CustomChannel1—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels. For example, community managers can use Experience Builder to determine where recommendations appear.
• CustomChannel2—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
• CustomChannel3—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
• CustomChannel4—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
• CustomChannel5—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
• DefaultChannel—Default recommendation channel. Recommendations appear by default on the Home and Question Detail pages of Customer Service and Partner Central Experience Builder templates. They also appear in the feed in the Salesforce mobile web and anywhere community managers add recommendations using Experience Builder.

maxResults
Type: Integer
Maximum number of recommendation results; default is 10. Values must be from 1 to 99.

Return Value
Type: ConnectApi.RecommendationCollection

Usage
If you want to get recommendations based on a recent action performed, such as following a user, use contextAction and contextObjectId together. For example, if you just followed Pam, you specify follow for contextAction and Pam’s user ID for contextObjectId.

This method only recommends users who are followed by people who follow Pam. In this example, John follows Pam so the method returns a recommendation for you to follow Suzanne since John also follows Suzanne.
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- `setTestGetRecommendationsForUser(communityId, userId, action, objectCategory, contextAction, contextObjectId, channel, maxResults, result)`
  
  *Apex Developer Guide: Testing ConnectApi Code*

---

**getScheduledRecommendation(communityId, scheduledRecommendationId)**

Get a scheduled custom recommendation.

**API Version**

35.0

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.ScheduledRecommendation getScheduledRecommendation(String communityId, String scheduledRecommendationId)
```

**Parameters**

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or `null`.

- **scheduledRecommendationId**
  - Type: String
  - ID of the scheduled custom recommendation.

**Return Value**

Type: `ConnectApi.ScheduledRecommendation`

**Usage**

Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

---

**getScheduledRecommendations(communityId, channel)**

Get scheduled custom recommendations.
API Version
36.0

Requires Chatter
Yes

Signature
public static ConnectApi.ScheduledRecommendationPage getScheduledRecommendations(String communityId, ConnectApi.RecommendationChannel channel)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.
channel
Type: ConnectApi.RecommendationChannel
A way to tie custom recommendations together. For example, display recommendations in specific places in the UI or show recommendations based on time of day or geographic locations. Values are:
• CustomChannel1—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels. For example, community managers can use Experience Builder to determine where recommendations appear.
• CustomChannel2—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
• CustomChannel3—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
• CustomChannel4—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
• CustomChannel5—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
• DefaultChannel—Default recommendation channel. Recommendations appear by default on the Home and Question Detail pages of Customer Service and Partner Central Experience Builder templates. They also appear in the feed in the Salesforce mobile web and anywhere community managers add recommendations using Experience Builder.

Return Value
Type: ConnectApi.ScheduledRecommendationPage

Usage
Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.
rejectRecommendationForUser(communityId, userId, action, objectId)
Reject a Chatter, custom, or static recommendation for the context user for the specified action and object ID.

API Version
33.0

Requires Chatter
Yes

Signature
public static rejectRecommendationForUser(String communityId, String userId, ConnectApi.RecommendationActionType action, String objectId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

userId
Type: String
ID for the context user or the keyword me.

action
Type: ConnectApi.RecommendationActionType
Specifies the action to take on a recommendation. Supported values are:
• follow—Follow a file, record, topic, or user.
• join—Join a group.
• view—View a file, group, article, record, user, custom, or static recommendation.

objectId
Type: String
Specifies the object to take action on.
• If action is follow, objectId is a user ID, file ID, record ID, or topic ID (version 36.0 and later).
• If action is join, objectId is a group ID.
• If action is view, objectId is a custom recommendation ID, the enum Today for static recommendations, or an article ID (version 37.0 and later).

Return Value
Type: Void

rejectRecommendationForUser(communityId, userId, action, objectEnum)
Reject a static recommendation for the context user.
API Version
34.0

Requires Chatter
Yes

Signature

```java
public static rejectRecommendationForUser(String communityId, String userId,
ConnectApi.RecommendationActionType action, ConnectApi.RecommendedObjectType objectEnum)
```

Parameters

- **communityId**
  Type: String
  ID for an Experience Cloud site, internal, or `null`.
- **userId**
  Type: String
  ID for the context user or the keyword `me`.
- **action**
  Type: ConnectApi.RecommendationActionType
  Specifies the action to take on a recommendation. Supported values are:
  - `view`—View a static recommendation.
- **objectEnum**
  Type: ConnectApi.RecommendedObjectType
  Specifies the object type to take action on.
  - `Today`—Static recommendations that don't have an ID, for example, the Today app recommendation.

Return Value

Type: Void

```java
updateRecommendationAudience(communityId, recommendationAudienceId, recommendationAudience)
```

Update a custom recommendation audience.

API Version
35.0

Requires Chatter
Yes
Signature

```java
public static ConnectApi.RecommendationAudience updateRecommendationAudience(String communityId, String recommendationAudienceId, ConnectApi.RecommendationAudienceInput recommendationAudience)
```

Parameters

- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- **recommendationAudienceId**
  - Type: `String`
  - ID of the custom recommendation audience.

- **recommendationAudience**
  - Type: `ConnectApi.RecommendationAudienceInput`

Return Value

- Type: `ConnectApi.RecommendationAudience`

Usage

Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

```java
updateRecommendationDefinition(communityId, recommendationDefinitionId, name, title, actionUrl, actionUrlName, explanation)
```

Update a custom recommendation definition with the specified parameters.

API Version

35.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.RecommendationDefinition updateRecommendationDefinition(String communityId, String recommendationDefinitionId, String name, String title, String actionUrl, String actionUrlName, String explanation recommendationDefinition)
```
Parameters

`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

`recommendationDefinitionId`
Type: `String`
ID of the custom recommendation definition.

`name`
Type: `String`
Name of the custom recommendation definition. The name is displayed in Setup.

`title`
Type: `String`
Title of the custom recommendation definition.

`actionUrl`
Type: `String`
URL for acting on the custom recommendation, for example, the URL to join a group.

`actionUrlName`
Type: `String`
Text label for the action URL in the user interface, for example, “Launch.”

`explanation`
Type: `String`
Explanation, or body, of the custom recommendation.

Return Value

Type: `ConnectApi.RecommendationDefinition`

Usage

Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

`updateRecommendationDefinition(communityId, recommendationDefinitionId, recommendationDefinition)`
Update a custom recommendation definition.

API Version

35.0
Requires Chatter
Yes

Signature
public static ConnectApi.RecommendationDefinition updateRecommendationDefinition(String communityId, String recommendationDefinitionId, ConnectApi.RecommendationDefinitionInput recommendationDefinition)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

recommendationDefinitionId
Type: String
ID of the custom recommendation definition.

recommendationDefinition
Type: ConnectApi.RecommendationDefinitionInput
A ConnectApi.RecommendationDefinitionInput object containing the properties to update.

Return Value
Type: ConnectApi.RecommendationDefinition

Usage
Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

updateRecommendationDefinitionPhoto(communityId, recommendationDefinitionId, fileUpload)
Update a custom recommendation definition photo with a file that hasn't been uploaded.

API Version
35.0

Requires Chatter
Yes
public static ConnectApi.Photo updateRecommendationDefinitionPhoto(String communityId, String recommendationDefinitionId, ConnectApi.BinaryInput fileUpload)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

recommendationDefinitionId
Type: String
ID of the custom recommendation definition.

fileUpload
Type: ConnectApi.BinaryInput
File to use as the photo. The content type must be usable as an image.

Return Value
Type: ConnectApi.Photo

Usage
Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

updateRecommendationDefinitionPhoto(communityId, recommendationDefinitionId, fileId, versionNumber)
Update a custom recommendation definition photo with an uploaded file.

API Version
35.0

Requires Chatter
Yes

Signature
public static ConnectApi.Photo updateRecommendationDefinitionPhoto(String communityId, String recommendationDefinitionId, String fileId, Integer versionNumber)
Parameters

communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

recommendationDefinitionId
  Type: String
  ID of the custom recommendation definition.

fileId
  Type: String
  ID of a file already uploaded. The file must be an image, and be smaller than 2 GB.

versionNumber
  Type: Integer
  Version number of the existing file. Specify either an existing version number, or null to get the latest version.

Return Value

Type: ConnectApi.Photo

Usage

Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

updateRecommendationDefinitionPhotoWithAttributes(communityId, recommendationDefinitionId, photo)

Update a custom recommendation definition photo with an uploaded file that requires cropping.

API Version

35.0

Requires Chatter

Yes

Signature

public static ConnectApi.Photo updateRecommendationDefinitionPhotoWithAttributes(String communityId, String recommendationDefinitionId, ConnectApi.PhotoInput photo)
ID for an Experience Cloud site, internal, or null.

**recommendationDefinitionId**
Type: **String**
ID of the custom recommendation definition.

**photo**
Type: **ConnectApi.PhotoInput**
A `ConnectApi.PhotoInput` object specifying the file ID, version number, and cropping parameters.

**Return Value**
Type: **ConnectApi.Photo**

**Usage**
Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

**updateRecommendationDefinitionPhotoWithAttributes(communityId, recommendationDefinitionId, photo, fileUpload)**
Update a custom recommendation definition photo with a file that hasn’t been uploaded and requires cropping.

**API Version**
35.0

**Requires Chatter**
Yes

**Signature**
```java
public static ConnectApi.Photo updateRecommendationDefinitionPhotoWithAttributes(String
communityId, String recommendationDefinitionId, ConnectApi.PhotoInput photo,
ConnectApi.BinaryInput fileUpload)
```

**Parameters**

**communityId**
Type: **String**
ID for an Experience Cloud site, internal, or null.

**recommendationDefinitionId**
Type: **String**
ID of the custom recommendation definition.
photo
   Type: ConnectApi.PhotoInput
   A ConnectApi.PhotoInput object specifying the cropping parameters.
fileUpload
   Type: ConnectApi.BinaryInput
   File to use as the photo. The content type must be usable as an image.

Return Value
Type: ConnectApi.Photo

Usage
Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

updateScheduledRecommendation(communityId, scheduledRecommendationId, scheduledRecommendation)
Update a scheduled custom recommendation.

API Version
35.0

Requires Chatter
Yes

Signature
public static ConnectApi.ScheduledRecommendation updateScheduledRecommendation(String communityId, String scheduledRecommendationId, ConnectApi.ScheduledRecommendationInput scheduledRecommendation)

Parameters
communityId
   Type: String
   ID for an Experience Cloud site, internal, or null.
scheduledRecommendationId
   Type: String
   ID of the scheduled custom recommendation.
scheduledRecommendation
   Type: ConnectApi.ScheduledRecommendationInput
A `ConnectApi.ScheduledRecommendationInput` object containing the properties to update.

**Return Value**

Type: `ConnectApi.ScheduledRecommendation`

**Usage**

Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

**Ranking scheduled custom recommendations example**

If you have these scheduled custom recommendations:

<table>
<thead>
<tr>
<th>Scheduled Recommendations</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>ScheduledRecommendationA</td>
<td>1</td>
</tr>
<tr>
<td>ScheduledRecommendationB</td>
<td>2</td>
</tr>
<tr>
<td>ScheduledRecommendationC</td>
<td>3</td>
</tr>
</tbody>
</table>

And you include this information in the Scheduled Custom Recommendation Input:

<table>
<thead>
<tr>
<th>Scheduled Recommendation</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>ScheduledRecommendationD</td>
<td>2</td>
</tr>
</tbody>
</table>

The result is:

<table>
<thead>
<tr>
<th>Scheduled Recommendation</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>ScheduledRecommendationA</td>
<td>1</td>
</tr>
<tr>
<td>ScheduledRecommendationD</td>
<td>2</td>
</tr>
<tr>
<td>ScheduledRecommendationB</td>
<td>3</td>
</tr>
<tr>
<td>ScheduledRecommendationC</td>
<td>4</td>
</tr>
</tbody>
</table>

**updateScheduledRecommendation(communityId, scheduledRecommendationId, rank, enabled, recommendationAudienceId)**

Update a scheduled custom recommendation with the specified parameters.

**API Version**

35.0
Requires Chatter
Yes

Signature

public static ConnectApi.ScheduledRecommendation updateScheduledRecommendation(String communityId, String scheduledRecommendationId, Integer rank, Boolean enabled, String recommendationAudienceId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

scheduledRecommendationId
Type: String
ID of the scheduled custom recommendation.

rank
Type: Integer
Relative rank of the scheduled custom recommendation indicated by ascending whole numbers starting with 1.
Setting the rank is comparable to an insertion into an ordered list. The scheduled custom recommendation is inserted into the position specified by the rank. The rank of all the scheduled custom recommendations after it is pushed down. See Ranking scheduled custom recommendations example.
If the specified rank is larger than the size of the list, the scheduled custom recommendation is put at the end of the list. The rank of the scheduled custom recommendation is the size of the list, instead of the one specified.
If a rank is not specified, the scheduled custom recommendation is put at the end of the list.

enabled
Type: Boolean
Indicates whether scheduling is enabled. If true, the custom recommendation is enabled and appears in Experience Cloud sites. If false, custom recommendations in feeds in Salesforce mobile web aren’t removed, but no new custom recommendations appear. In Customer Service and Partner Central sites, disabled custom recommendations no longer appear.

recommendationAudienceId
Type: String
ID of the custom recommendation definition that this scheduled recommendation schedules.

Return Value
Type: ConnectApi.ScheduledRecommendation

Usage
Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.
Ranking scheduled custom recommendations example

If you have these scheduled custom recommendations:

<table>
<thead>
<tr>
<th>Scheduled Recommendations</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>ScheduledRecommendationA</td>
<td>1</td>
</tr>
<tr>
<td>ScheduledRecommendationB</td>
<td>2</td>
</tr>
<tr>
<td>ScheduledRecommendationC</td>
<td>3</td>
</tr>
</tbody>
</table>

And you include this information in the Scheduled Custom Recommendation Input:

<table>
<thead>
<tr>
<th>Scheduled Recommendation</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>ScheduledRecommendationD</td>
<td>2</td>
</tr>
</tbody>
</table>

The result is:

<table>
<thead>
<tr>
<th>Scheduled Recommendation</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>ScheduledRecommendationA</td>
<td>1</td>
</tr>
<tr>
<td>ScheduledRecommendationD</td>
<td>2</td>
</tr>
<tr>
<td>ScheduledRecommendationB</td>
<td>3</td>
</tr>
<tr>
<td>ScheduledRecommendationC</td>
<td>4</td>
</tr>
</tbody>
</table>

Recommendations Test Methods

The following are the test methods for Recommendations. All methods are static.

For information about using these methods to test your ConnectApi code, see Testing ConnectApi Code.

IN THIS SECTION:

`setTestGetRecommendationForUser(communityId, userId, action, objectId, result)`
Register a `ConnectApi.RecommendationCollection` object to be returned when `getRecommendationForUser` is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

`setTestGetRecommendationsForUser(communityId, userId, contextAction, contextObjectId, channel, maxResults, result)`
Register a `ConnectApi.RecommendationCollection` object to be returned when `getRecommendationsForUser` is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

`setTestGetRecommendationsForUser(communityId, userId, action, contextAction, contextObjectId, channel, maxResults, result)`
Register a `ConnectApi.RecommendationCollection` object to be returned when `getRecommendationsForUser` is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.
setTestGetRecommendationsForUser(communityId, userId, action, objectCategory, contextAction, contextObjectId, channel, maxResults, result)

Register a ConnectApi.RecommendationCollection object to be returned when getRecommendationsForUser is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

setTestGetRecommendationForUser(communityId, userId, action, objectId, result)

Register a ConnectApi.RecommendationCollection object to be returned when getRecommendationForUser is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

API Version
33.0

Requires Chatter
Yes

Signature
public static Void setTestGetRecommendationForUser(String communityId, String userId, ConnectApi.RecommendationActionType action, String objectId, ConnectApi.RecommendationCollection result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

userId
Type: String
ID for the context user or the keyword me.

action
Type: ConnectApi.RecommendationActionType
Specifies the action to take on a recommendation.
• follow—Follow a file, record, topic, or user.
• join—Join a group.
• view—View a file, group, article, record, user, custom, or static recommendation.

objectId
Type: String
Specifies the object to take action on.
• If action is follow, objectId is a user ID, file ID, record ID, or topic ID (version 36.0 and later).
• If action is join, objectId is a group ID.
• If action is view, objectId is a user ID, file ID, group ID, record ID, custom recommendation ID, the enum Today for static recommendations, or an article ID (version 37.0 and later).
result
    Type: `ConnectApi.RecommendationCollection`
    Object containing test data.

Return Value
Type: Void

SEE ALSO:
    getRecommendationForUser(communityId, userId, action, objectId)
    *Apex Developer Guide: Testing ConnectApi Code*

`setTestGetRecommendationsForUser(communityId, userId, contextAction, contextObjectId, channel, maxResults, result)`
Register a `ConnectApi.RecommendationCollection` object to be returned when `getRecommendationsForUser` is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

API Version
36.0

Requires Chatter
Yes

Signature
```
public static Void setTestGetRecommendationsForUser(String communityId, String userId, 
    ConnectApi.RecommendationActionType contextAction, String contextObjectId, 
    ConnectApi.RecommendationChannel channel, Integer maxResults, 
    ConnectApi.RecommendationCollection result)
```

Parameters

`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

`userId`
Type: `String`
ID for the context user or the keyword `me`.

`contextAction`
Type: `ConnectApi.RecommendationActionType`
Action that the context user just performed. Supported values are:
- `follow`
- `view`
Use `contextAction` and `contextObjectId` together to get new recommendations based on the action just performed. If you don’t want recommendations based on a recent action, specify `null`.

**contextObjectId**
Type: `String`
ID of the object that the context user just performed an action on.
- If `contextAction` is `follow`, `contextObjectId` is a user ID, file ID, record ID, or topic ID.
- If `contextAction` is `view`, `contextObjectId` is a user ID, file ID, group ID, record ID, or article ID (version 37.0 and later).

Use `contextAction` and `contextObjectId` together to get new recommendations based on the action just performed. If you don’t want recommendations based on a recent action, specify `null`.

**channel**
Type: `ConnectApi.RecommendationChannel`
A way to tie custom recommendations together. For example, display recommendations in specific places in the UI or show recommendations based on time of day or geographic locations. Values are:
- `CustomChannel1`—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels. For example, community managers can use Experience Builder to determine where recommendations appear.
- `CustomChannel2`—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
- `CustomChannel3`—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
- `CustomChannel4`—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
- `CustomChannel5`—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.

**maxResults**
Type: `Integer`
Maximum number of recommendation results; default is 10. Values must be from 1 to 99.

**result**
Type: `ConnectApi.RecommendationCollection`
Object containing test data.

**Return Value**
Type: `Void`

**SEE ALSO:**
- `getRecommendationsForUser(communityId, userId, contextAction, contextObjectId, channel, maxResults)`
- *Apex Developer Guide: Testing ConnectApi Code*
setTestGetRecommendationsForUser(communityId, userId, action, contextAction, contextObjectId, channel, maxResults, result)

Register a ConnectApi.RecommendationCollection object to be returned when getRecommendationsForUser is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

API Version
36.0

Requires Chatter
Yes

Signature
public static Void setTestGetRecommendationsForUser(String communityId, String userId, ConnectApi.RecommendationActionType action, ConnectApi.RecommendationActionType contextAction, String contextObjectId, ConnectApi.RecommendationChannel channel, Integer maxResults, ConnectApi.RecommendationCollection result)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

userId
Type: String
ID for the context user or the keyword me.

action
Type: ConnectApi.RecommendationActionType
Specifies the action to take on a recommendation.
• follow—Follow a file, record, topic, or user.
• join—Join a group.
• view—View a file, group, article, record, user, custom, or static recommendation.

contextAction
Type: ConnectApi.RecommendationActionType
Action that the context user just performed. Supported values are:
• follow
• view

Use contextAction and contextObjectId together to get new recommendations based on the action just performed. If you don’t want recommendations based on a recent action, specify null.

contextObjectId
Type: String
ID of the object that the context user just performed an action on.
- If `contextAction` is `follow`, `contextObjectId` is a user ID, file ID, record ID, or topic ID.
- If `contextAction` is `view`, `contextObjectId` is a user ID, file ID, group ID, record ID, or article ID (version 37.0 and later).

Use `contextAction` and `contextObjectId` together to get new recommendations based on the action just performed. If you don't want recommendations based on a recent action, specify `null`.

**channel**

Type: `ConnectApi.RecommendationChannel`

A way to tie custom recommendations together. For example, display recommendations in specific places in the UI or show recommendations based on time of day or geographic locations. Values are:

- **CustomChannel1**—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels. For example, community managers can use Experience Builder to determine where recommendations appear.
- **CustomChannel2**—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
- **CustomChannel3**—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
- **CustomChannel4**—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
- **CustomChannel5**—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.
- **DefaultChannel**—Default recommendation channel. Recommendations appear by default on the Home and Question Detail pages of Customer Service and Partner Central Experience Builder templates. They also appear in the feed in the Salesforce mobile web and anywhere community managers add recommendations using Experience Builder.

**maxResults**

Type: `Integer`

Maximum number of recommendation results; default is 10. Values must be from 1 to 99.

**result**

Type: `ConnectApi.RecommendationCollection`

Object containing test data.

**Return Value**

Type: Void

SEE ALSO:

- `getRecommendationsForUser(communityId, userId, action, contextAction, contextObjectId, channel, maxResults)`

- `setTestGetRecommendationsForUser(communityId, userId, action, objectCategory, contextAction, contextObjectId, channel, maxResults, result)`

Register a `ConnectApi.RecommendationCollection` object to be returned when `getRecommendationsForUser` is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.
API Version
36.0

Requires Chatter
Yes

Signature
public static Void setTestGetRecommendationsForUser(String communityId, String userId, ConnectApi.RecommendationActionType action, String objectCategory, ConnectApi.RecommendationActionType contextAction, String contextObjectId, ConnectApi.RecommendationChannel channel, Integer maxResults, ConnectApi.RecommendationCollection result)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

userId
Type: String
ID for the context user or the keyword me.

action
Type: ConnectApi.RecommendationActionType
Specifies the action to take on a recommendation.
• follow—Follow a file, record, topic, or user.
• join—Join a group.
• view—View a file, group, article, record, user, custom, or static recommendation.

objectCategory
Type: String
• If action is follow, objectCategory is users, files, records, or topics.
• If action is join, objectCategory is groups.
• If action is view, objectCategory is users, files, groups, records, custom, apps, or articles (version 37.0 and later).

You can also specify a key prefix, the first three characters of the object ID, as the objectCategory. Valid values are:
• If action is follow, objectCategory is 005 (users), 069 (files), 0TO (topics), or 001 (accounts), for example.
• If action is join, objectCategory is 0F9 (groups).
• If action is view, objectCategory is 005 (users), 069 (files), 0F9 (groups), 0RD (custom recommendations), T (static recommendations), 001 (accounts), or kA0 (articles), for example, (version 370 and later).

contextAction
Type: ConnectApi.RecommendationActionType
Action that the context user just performed. Supported values are:
• follow
• view

Use `contextAction` and `contextObjectId` together to get new recommendations based on the action just performed. If you don’t want recommendations based on a recent action, specify `null`.

`contextObjectId`  
Type: `String`  
ID of the object that the context user just performed an action on.  
- If `contextAction` is `follow`, `contextObjectId` is a user ID, file ID, record ID, or topic ID.  
- If `contextAction` is `view`, `contextObjectId` is a user ID, file ID, group ID, record ID, or article ID (version 37.0 and later).

Use `contextAction` and `contextObjectId` together to get new recommendations based on the action just performed. If you don’t want recommendations based on a recent action, specify `null`.

`channel`  
Type: `ConnectApi.RecommendationChannel`  
A way to tie custom recommendations together. For example, display recommendations in specific places in the UI or show recommendations based on time of day or geographic locations. Values are:  
- `CustomChannel1` — Custom recommendation channel. Not used by default. Work with your community manager to define custom channels. For example, community managers can use Experience Builder to determine where recommendations appear.  
- `CustomChannel2` — Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.  
- `CustomChannel3` — Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.  
- `CustomChannel4` — Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.  
- `CustomChannel5` — Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.  

`maxResults`  
Type: `Integer`  
Maximum number of recommendation results; default is 10. Values must be from 1 to 99.

`result`  
Type: `ConnectApi.RecommendationCollection`  
Object containing test data.

**Return Value**

Type: `Void`

SEE ALSO:  
`getRecommendationsForUser(communityId, userId, action, objectCategory, contextAction, contextObjectId, channel, maxResults)`  
*Apex Developer Guide: Testing ConnectApi Code*
## Retired Recommendations Methods

The following methods for Recommendations are retired.

### IN THIS SECTION:

- `createScheduledRecommendation(communityId, recommendationDefinitionId, rank, enabled, recommendationAudienceId)`: Create a scheduled custom recommendation with the specified parameters.
- `getRecommendationsForUser(communityId, userId, contextAction, contextObjectId, maxResults)`: Get the Chatter recommendations, such as user, group, file, and record recommendations for the context user. Get the custom and static recommendations for the context user.
- `getRecommendationsForUser(communityId, userId, action, contextAction, contextObjectId, maxResults)`: Get the Chatter, custom, and static recommendations for the context user for the specified action.
- `getRecommendationsForUser(communityId, userId, action, objectCategory, contextAction, contextObjectId, maxResults)`: Get the Chatter, custom, and static recommendations for the context user for the specified action and object category.
- `getScheduledRecommendations(communityId)`: Get scheduled custom recommendations.
- `setTestGetRecommendationsForUser(communityId, userId, contextAction, contextObjectId, maxResults, result)`: Register a `ConnectApi.RecommendationCollection` object to be returned when `getRecommendationsForUser` is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>createScheduledRecommendation</code></td>
<td>Create a scheduled custom recommendation with the specified parameters.</td>
</tr>
<tr>
<td><code>getRecommendationsForUser</code></td>
<td>Get the Chatter recommendations, such as user, group, file, and record recommendations for the context user. Get the custom and static recommendations for the context user.</td>
</tr>
<tr>
<td><code>getRecommendationsForUser(communityId, userId, action, contextAction, contextObjectId, maxResults)</code></td>
<td>Get the Chatter, custom, and static recommendations for the context user for the specified action.</td>
</tr>
<tr>
<td><code>setTestGetRecommendationsForUser</code></td>
<td>Register a <code>ConnectApi.RecommendationCollection</code> object to be returned when <code>getRecommendationsForUser</code> is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.</td>
</tr>
</tbody>
</table>

### API Version

35.0 only

**Important:** In version 36.0 and later, use `createScheduledRecommendation(communityId, recommendationDefinitionId, rank, enabled, recommendationAudienceId, channel)`.

### Requires Chatter

Yes
Signature

public static ConnectApi.ScheduledRecommendation createScheduledRecommendation(String communityId, String recommendationDefinitionId, Integer rank, Boolean enabled, String recommendationAudienceId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

recommendationDefinitionId
Type: String
ID of the custom recommendation definition.

rank
Type: Integer
Relative rank of the scheduled custom recommendation indicated by ascending whole numbers starting with 1.
Setting the rank is comparable to an insertion into an ordered list. The scheduled custom recommendation is inserted into the position specified by the rank. The rank of all the scheduled custom recommendations after it is pushed down. See Ranking scheduled custom recommendations example.
If the specified rank is larger than the size of the list, the scheduled custom recommendation is put at the end of the list. The rank of the scheduled custom recommendation is the size of the list, instead of the one specified.
If a rank is not specified, the scheduled custom recommendation is put at the end of the list.

enabled
Type: Boolean
Indicates whether scheduling is enabled. If true, the custom recommendation is enabled and appears in Experience Cloud sites. If false, custom recommendations in feeds in Salesforce mobile web aren’t removed, but no new custom recommendations appear. In Customer Service and Partner Central sites, disabled custom recommendations no longer appear.

recommendationAudienceId
Type: String
ID of the custom recommendation definition that this scheduled recommendation schedules.

Return Value

Type: ConnectApi.ScheduledRecommendation

Usage

Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

Ranking scheduled custom recommendations example

If you have these scheduled custom recommendations:
And you include this information in the Scheduled Custom Recommendation Input:

<table>
<thead>
<tr>
<th>Scheduled Recommendation</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>ScheduledRecommendationA</td>
<td>1</td>
</tr>
<tr>
<td>ScheduledRecommendationB</td>
<td>2</td>
</tr>
<tr>
<td>ScheduledRecommendationC</td>
<td>3</td>
</tr>
</tbody>
</table>

The result is:

<table>
<thead>
<tr>
<th>Scheduled Recommendation</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>ScheduledRecommendationA</td>
<td>1</td>
</tr>
<tr>
<td>ScheduledRecommendationD</td>
<td>2</td>
</tr>
<tr>
<td>ScheduledRecommendationB</td>
<td>3</td>
</tr>
<tr>
<td>ScheduledRecommendationC</td>
<td>4</td>
</tr>
</tbody>
</table>

getRecommendationsForUser(communityId, userId, contextAction, contextObjectId, maxResults)

Get the Chatter recommendations, such as user, group, file, and record recommendations for the context user. Get the custom and static recommendations for the context user.

API Version

33.0–35.0

Important: In version 36.0 and later, use getRecommendationsForUser(communityId, userId, contextAction, contextObjectId, channel, maxResults).

Requires Chatter

Yes

Signature

public static ConnectApi.RecommendationCollection getRecommendationsForUser(String communityId, String userId, ConnectApi.RecommendationActionType contextAction, String contextObjectId, Integer maxResults)
Parameters

*communityId*
Type: String
ID for an Experience Cloud site, internal, or null.

*userId*
Type: String
ID for the context user or the keyword me.

*contextAction*
Type: ConnectApi.RecommendationActionType
Action that the context user just performed. Supported values are:
- follow
- view

Use *contextAction* and *contextObjectId* together to get new recommendations based on the action just performed.
If you don’t want recommendations based on a recent action, specify null.

*contextObjectId*
Type: String
ID of the object that the context user just performed an action on.
- If *contextAction* is follow, *contextObjectId* is a user ID, file ID, or record ID.
- If *contextAction* is view, *contextObjectId* is a user ID, file ID, group ID, or record ID.

Use *contextAction* and *contextObjectId* together to get new recommendations based on the action just performed.
If you don’t want recommendations based on a recent action, specify null.

*maxResults*
Type: Integer
Maximum number of recommendation results; default is 10. Values must be from 1 to 99.

Return Value

Type: ConnectApi.RecommendationCollection

Usage

If you want to get recommendations based on a recent action performed, such as following a user, use *contextAction* and *contextObjectId* together. For example, if you just followed Pam, you specify follow for *contextAction* and Pam’s user ID for *contextObjectId*.

This method only recommends users who are followed by people who follow Pam. In this example, John follows Pam so the method returns a recommendation for you to follow Suzanne since John also follows Suzanne.

To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

* setTestGetRecommendationsForUser(communityId, userId, contextAction, contextObjectId, maxResults, result)

1440
getRecommendationsForUser(communityId, userId, action, contextAction, contextObjectId, maxResults)

Get the Chatter, custom, and static recommendations for the context user for the specified action.

API Version

33.0–35.0

⚠️ Important: In version 36.0 and later, use getRecommendationsForUser(communityId, userId, action, contextAction, contextObjectId, channel, maxResults).

Requires Chatter

Yes

Signature

public static ConnectApi.RecommendationCollection getRecommendationsForUser(String communityId, String userId, ConnectApi.RecommendationActionType action, ConnectApi.RecommendationActionType contextAction, String contextObjectId, Integer maxResults)

Parameters

communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

userId
  Type: String
  ID for the context user or the keyword me.

action
  Type: ConnectApi.RecommendationActionType
  Specifies the action to take on a recommendation.
  • follow—Follow a file, record, topic, or user.
  • join—Join a group.
  • view—View a file, group, article, record, user, custom, or static recommendation.

contextAction
  Type: ConnectApi.RecommendationActionType
  Action that the context user just performed. Supported values are:
  • follow
  • view
  Use contextAction and contextObjectId together to get new recommendations based on the action just performed. If you don’t want recommendations based on a recent action, specify null.

contextObjectId
  Type: String
ID of the object that the context user just performed an action on.

- If `contextAction` is `follow`, `contextObjectId` is a user ID, file ID, or record ID.
- If `contextAction` is `view`, `contextObjectId` is a user ID, file ID, group ID, or record ID.

Use `contextAction` and `contextObjectId` together to get new recommendations based on the action just performed. If you don’t want recommendations based on a recent action, specify `null`.

`maxResults`

  
  Type: `Integer`

  Maximum number of recommendation results; default is 10. Values must be from 1 to 99.

**Return Value**

Type: `ConnectApi.RecommendationCollection`

**Usage**

If you want to get recommendations based on a recent action performed, such as following a user, use `contextAction` and `contextObjectId` together. For example, if you just followed Pam, you specify `follow` for `contextAction` and Pam’s user ID for `contextObjectId`.

This method only recommends users who are followed by people who follow Pam. In this example, John follows Pam so the method returns a recommendation for you to follow Suzanne since John also follows Suzanne.

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

**SEE ALSO:**

- `setTestGetRecommendationsForUser(communityId, userId, action, contextAction, contextObjectId, maxResults, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

```
getRecommendationsForUser(communityId, userId, action, objectCategory, contextAction, contextObjectId, maxResults)
```

Get the Chatter, custom, and static recommendations for the context user for the specified action and object category.

**API Version**

33.0–35.0

**Important:** In version 36.0 and later, use `getRecommendationsForUser(communityId, userId, action, objectCategory, contextAction, contextObjectId, channel, maxResults)`.

**Requires Chatter**

Yes

**Signature**

```
public static ConnectApi.RecommendationCollection getRecommendationsForUser(String communityId, String userId, ConnectApi.RecommendationActionType action, String
```
objectCategory, ConnectApi.RecommendationActionType contextAction, String
contextObjectId, Integer maxResults)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

userId
Type: String
ID for the context user or the keyword me.

action
Type: ConnectApi.RecommendationActionType
Specifies the action to take on a recommendation.
- follow—Follow a file, record, topic, or user.
- join—Join a group.
- view—View a file, group, article, record, user, custom, or static recommendation.

objectCategory
Type: String
- If action is follow, objectCategory is users, files, or records.
- If action is join, objectCategory is groups.
- If action is view, objectCategory is users, files, groups, records, custom, or apps.
You can also specify a key prefix, the first three characters of the object ID, as the objectCategory. Valid values are:
- If action is follow, objectCategory is 005 (users), 069 (files), or 001 (accounts), for example.
- If action is join, objectCategory is 0F9 (groups).
- If action is view, objectCategory is 005 (users), 069 (files), 0F9 (groups), 0RD (custom recommendations), T (static recommendations), or 001 (accounts), for example.

customAction
Type: ConnectApi.RecommendationActionType
Action that the context user just performed. Supported values are:
- follow
- view

Use contextAction and contextObjectId together to get new recommendations based on the action just performed. If you don’t want recommendations based on a recent action, specify null.

customObjectId
Type: String
ID of the object that the context user just performed an action on.
- If contextAction is follow, contextObjectId is a user ID, file ID, or record ID.
- If contextAction is view, contextObjectId is a user ID, file ID, group ID, or record ID.

Use contextAction and contextObjectId together to get new recommendations based on the action just performed. If you don’t want recommendations based on a recent action, specify null.
**maxResults**

Type: `Integer`

Maximum number of recommendation results; default is 10. Values must be from 1 to 99.

**Return Value**

Type: `ConnectApi.RecommendationCollection`

**Usage**

If you want to get recommendations based on a recent action performed, such as following a user, use `contextAction` and `contextObjectId` together. For example, if you just followed Pam, you specify `follow` for `contextAction` and Pam’s user ID for `contextObjectId`.

This method only recommends users who are followed by people who follow Pam. In this example, John follows Pam so the method returns a recommendation for you to follow Suzanne since John also follows Suzanne.

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

**SEE ALSO:**

- `setTestGetRecommendationsForUser(communityId, userId, action, objectCategory, contextAction, contextObjectId, maxResults, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

**getScheduledRecommendations(communityId)**

Get scheduled custom recommendations.

**API Version**

35.0 only
Important: In version 36.0 and later, use `getScheduledRecommendations(communityId, channel)`.

Requires Chatter
Yes

Signature

```java
public static ConnectApi.ScheduledRecommendationPage getScheduledRecommendations(String communityId)
```

Parameters

- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.

Return Value

Type: `ConnectApi.ScheduledRecommendationPage`

Usage

Community managers can access, create, and delete audiences, definitions, and schedules for custom recommendations. (Community managers are users with the Create and Set Up Experiences or Manage Experiences permission.) Users with the Modify All Data permission can also access, create, and delete custom recommendation audiences, custom recommendation definitions, and scheduled custom recommendations.

```java
setTestGetRecommendationsForUser(communityId, userId, contextAction, contextObjectId, maxResults, result)
```

Register a `ConnectApi.RecommendationCollection` object to be returned when `getRecommendationsForUser` is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

API Version

33.0–35.0

Requires Chatter
Yes

Signature

```java
public static Void setTestGetRecommendationsForUser(String communityId, String userId, ConnectApi.RecommendationActionType contextAction, String contextObjectId, Integer maxResults, ConnectApi.RecommendationCollection result)
```
Parameters

`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

`userId`
Type: `String`
ID for the context user or the keyword `me`.

`contextAction`
Type: `ConnectApi.RecommendationActionType`
Action that the context user just performed. Supported values are:
- `follow`
- `view`

Use `contextAction` and `contextObjectId` together to get new recommendations based on the action just performed. If you don’t want recommendations based on a recent action, specify `null`.

`contextObjectId`
Type: `String`
ID of the object that the context user just performed an action on.
- If `contextAction` is `follow`, `contextObjectId` is a user ID, file ID, or record ID.
- If `contextAction` is `view`, `contextObjectId` is a user ID, file ID, group ID, or record ID.

Use `contextAction` and `contextObjectId` together to get new recommendations based on the action just performed. If you don’t want recommendations based on a recent action, specify `null`.

`maxResults`
Type: `Integer`
Maximum number of recommendation results; default is 10. Values must be from 1 to 99.

`result`
Type: `ConnectApi.RecommendationCollection`
Object containing test data.

Return Value
Type: Void

SEE ALSO:
- `getRecommendationsForUser(communityId, userId, contextAction, contextObjectId, maxResults)`

`setTestGetRecommendationsForUser(communityId, userId, action, contextAction, contextObjectId, maxResults, result)`
Register a `ConnectApi.RecommendationCollection` object to be returned when `getRecommendationsForUser` is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.
API Version

33.0–35.0

Requires Chatter

Yes

Signature

public static Void setTestGetRecommendationsForUser(String communityId, String userId, ConnectApi.RecommendationActionType action, ConnectApi.RecommendationActionType contextAction, String contextObjectId, Integer maxResults, ConnectApi.RecommendationCollection result)

Parameters

communityId

Type: String

ID for an Experience Cloud site, internal, or null.

userId

Type: String

ID for the context user or the keyword me.

action

Type: ConnectApi.RecommendationActionType

Specifies the action to take on a recommendation.
  • follow—Follow a file, record, topic, or user.
  • join—Join a group.
  • view—View a file, group, article, record, user, custom, or static recommendation.

contextAction

Type: ConnectApi.RecommendationActionType

Action that the context user just performed. Supported values are:
  • follow
  • view

Use contextAction and contextObjectId together to get new recommendations based on the action just performed. If you don’t want recommendations based on a recent action, specify null.

contextObjectId

Type: String

ID of the object that the context user just performed an action on.
  • If contextAction is follow, contextObjectId is a user ID, file ID, or record ID.
  • If contextAction is view, contextObjectId is a user ID, file ID, group ID, or record ID.

Use contextAction and contextObjectId together to get new recommendations based on the action just performed. If you don’t want recommendations based on a recent action, specify null.
maxResults
  Type: Integer
  Maximum number of recommendation results; default is 10. Values must be from 1 to 99.

result
  Type: ConnectApi.RecommendationCollection
  Object containing test data.

Return Value
  Type: Void

SEE ALSO:
  getRecommendationsForUser(communityId, userId, action, contextAction, contextObjectId, maxResults)

setTestGetRecommendationsForUser(communityId, userId, action, objectCategory, contextAction, contextObjectId, maxResults, result)

Register a ConnectApi.RecommendationCollection object to be returned when getRecommendationsForUser is called with matching parameters in a test context. Use the method with the same parameters or the code throws an exception.

API Version
  33.0–35.0

Requires Chatter
  Yes

Signature
  public static Void setTestGetRecommendationsForUser(String communityId, String userId, ConnectApi.RecommendationActionType action, String objectCategory, ConnectApi.RecommendationActionType contextAction, String contextObjectId, Integer maxResults, ConnectApi.RecommendationCollection result)

Parameters
  communityId
    Type: String
    ID for an Experience Cloud site, internal, or null.
  userId
    Type: String
    ID for the context user or the keyword me.
  action
    Type: ConnectApi.RecommendationActionType
Specifies the action to take on a recommendation.
- **follow**—Follow a file, record, topic, or user.
- **join**—Join a group.
- **view**—View a file, group, article, record, user, custom, or static recommendation.

**objectCategory**
Type: String
- If **action** is **follow**, **objectCategory** is users, files, or records.
- If **action** is **join**, **objectCategory** is groups.
- If **action** is **view**, **objectCategory** is users, files, groups, records, custom, or apps.

You can also specify a key prefix, the first three characters of the object ID, as the **objectCategory**. Valid values are:
- If **action** is **follow**, **objectCategory** is 005 (users), 069 (files), or 001 (accounts), for example.
- If **action** is **join**, **objectCategory** is 0F9 (groups).
- If **action** is **view**, **objectCategory** is 005 (users), 069 (files), 0F9 (groups), 0RD (custom recommendations), T (static recommendations), or 001 (accounts), for example.

**contextAction**
Type: ConnectApi.RecommendationActionType
Action that the context user just performed. Supported values are:
- **follow**
- **view**

Use **contextAction** and **contextObjectId** together to get new recommendations based on the action just performed. If you don’t want recommendations based on a recent action, specify null.

**contextObjectId**
Type: String
ID of the object that the context user just performed an action on.
- If **contextAction** is **follow**, **contextObjectId** is a user ID, file ID, or record ID.
- If **contextAction** is **view**, **contextObjectId** is a user ID, file ID, group ID, or record ID.

Use **contextAction** and **contextObjectId** together to get new recommendations based on the action just performed. If you don’t want recommendations based on a recent action, specify null.

**maxResults**
Type: Integer
Maximum number of recommendation results; default is 10. Values must be from 1 to 99.

**result**
Type: ConnectApi.RecommendationCollection
Object containing test data.
Records Class
Access information about record motifs, which are small icons used to distinguish record types in the Salesforce UI.

Namespace
ConnectApi

Records Methods
The following are methods for Records. All methods are static.

IN THIS SECTION:
- getMotif(communityId, idOrPrefix)
  Get a motif that contains the URLs for a set of small, medium, and large motif icons for a record. It can also contain a base color for the record.
- getMotifBatch(communityId, idOrPrefixList)
  Get a motif for a list of objects.

getMotif(communityId, idOrPrefix)
Get a motif that contains the URLs for a set of small, medium, and large motif icons for a record. It can also contain a base color for the record.

API Version
28.0

Requires Chatter
No

Signature
```
public static ConnectApi.Motif getMotif(String communityId, String idOrPrefix)
```

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

$idOrPrefix
Type: String
An ID or key prefix.

Return Value
Type: ConnectApi.Motif

Usage
Each Salesforce record type has its own set of motif icons.

getMotifBatch(communityId, idOrPrefixList)
Get a motif for a list of objects.

API Version
31.0

Requires Chatter
No

Signature
public static ConnectApi.BatchResult[] getMotifBatch(String communityId, List<String> idOrPrefixList)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

idOrPrefixList
Type: List<String>
A list of object IDs or prefixes.

Return Value
Type: ConnectApi.BatchResult[]
Example

```java
String communityId = null;
List<String> prefixIds = new List<String> { '001', '01Z', '069' };

// Get info about the motifs of all records in the list.
ConnectApi.BatchResult[] batchResults = ConnectApi.Records.getMotifBatch(communityId, prefixIds);

for (ConnectApi.BatchResult batchResult : batchResults) {
    if (batchResult.isSuccess()) {
        // Operation was successful.
        // Print the color of each motif.
        ConnectApi.Motif motif;
        if (batchResult.getResult() instanceof ConnectApi.Motif) {
            motif = (ConnectApi.Motif) batchResult.getResult();
        }
        System.debug('SUCCESS');
        System.debug(motif.color);
    }
    else {
        // Operation failed. Print errors.
        System.debug('FAILURE');
        System.debug(batchResult.getErrorMessage());
    }
}
```

Repricing Class

Perform functions related to repricing orders in Order Management.

Namespace

ConnectApi

Repricing Methods

The following are methods for Repricing. All methods are static.

IN THIS SECTION:

- `productDetails(webstoreId, skuOrProductId, effectiveAccountId, currencyCode, locale)`
  Get details of a product in a web store.

- `searchProducts(webstoreId, searchTerm, pageParam, pageSize, effectiveAccountId, facets)`
  Search products in a webstore.

- `productDetails(webstoreId, skuOrProductId, effectiveAccountId, currencyCode, locale)`
  Get details of a product in a web store.
API Version
55.0

Requires Chatter
No

Signature
public static ConnectApi.ProductDetailsOutputRepresentation productDetails(String webstoreId, String skuOrProductId, String effectiveAccountId, String currencyCode, String locale)

Parameters

webstoreId
Type: String
ID of the WebStore.

skuOrProductId
Type: String
SKU or ID of the Product.

effectiveAccountId
Type: String
Effective Account ID. Required for B2B stores. For other stores, pass null.

currencyCode
Type: String
ISO currency code. If you pass null, the default store value is used.

locale
Type: String
Locale. If you pass null, the default store value is used.

excludeAttributeSetInfo
Type: String
Specifies whether the attribute set information for the product is returned.

excludeMedia
Type: String
Specifies whether the media groups and default images of the product are returned.

excludeQuantityRule
Type: String
Specifies whether the quantity rule information for the product is returned.

excludeVariationInfo
Type: String
Specifies whether the variation information for the product is returned.
excludePrices
  Type: String
  Specifies whether the prices for the product is returned.

Return Value
Type: ConnectApi.ProductDetailsOutputRepresentation

searchProducts(webstoreId, searchTerm, pageParam, pageSize, effectiveAccountId, facets)
Search products in a webstore.

API Version
59.0

Requires Chatter
No

Signature
public static ConnectApi.ProductSearchOutputRepresentation searchProducts(String webstoreId, String searchTerm, Integer pageParam, Integer pageSize, String effectiveAccountId, String facets)

Parameters
webstoreId
  Type: String
  ID of the webstore.

searchTerm
  Type: String
  Term used for the search.

pageParam
  Type: Integer
  Maximum number of search results pages to return. If you don't specify a value, the default is 1.

pageSize
  Type: Integer
  Number of items per page. Valid values are from 1 through 100. If you don't specify a value, the default size is 20.

effectiveAccountId
  Type: String
  ID of the account for which the request is made. If unspecified, defaults to the account ID for the context user.

facets
  Type: String
A list of facet names to filter the search. For example, ["size_medium", "color_red"] is encoded to WyJzaXplX21lZGl1bSIsICJjb2xvcl9yZWQiXQ==

Return Value
Type: `ConnectApi.ProductSearchOutputRepresentation` on page 1922

ReturnOrder Class
Process ReturnOrders in Order Management.

Namespace
`ConnectApi`

ReturnOrder Methods
The following are methods for `ReturnOrder`. All methods are static.

IN THIS SECTION:
- `createReturnOrder(returnOrderInput)`
  - Create a ReturnOrder and ReturnOrderLineItems for items belonging to an OrderSummary.
- `returnItems(returnOrderId, returnItemsInput)`
  - Process ReturnOrderLineItems belonging to a ReturnOrder. Processing a ReturnOrderLineItem generates a change Order and makes that ReturnOrderLineItem read-only. The change order for a returned item or delivery charge has a positive amount and should be used to create a credit memo. The change order for a return fee has a negative amount and should be used to create an invoice. If a processed ReturnOrderLineItem has any remaining expected quantity, then the API creates a separate ReturnOrderLineItem representing that quantity.

`createReturnOrder(returnOrderInput)`
Create a ReturnOrder and ReturnOrderLineItems for items belonging to an OrderSummary.

API Version
50.0

Requires Chatter
No

Signature
```java
public static ConnectApi.ReturnOrderOutputRepresentation createReturnOrder(ConnectApi.ReturnOrderInputRepresentation returnOrderInput)
```
Parameters

`returnOrderInput`
Type: `ConnectApi.ReturnOrderInputRepresentation`
Data for creating a ReturnOrder and ReturnOrderLineItems.

Return Value

Type: `ConnectApi.ReturnOrderOutputRepresentation`

SEE ALSO:

`returnItems(returnOrderId, returnItemsInput)`

`returnItems(returnOrderId, returnItemsInput)`
Process ReturnOrderLineItems belonging to a ReturnOrder. Processing a ReturnOrderLineItem generates a change Order and makes that ReturnOrderLineItem read-only. The change order for a returned item or delivery charge has a positive amount and should be used to create a credit memo. The change order for a return fee has a negative amount and should be used to create an invoice. If a processed ReturnOrderLineItem has any remaining expected quantity, then the API creates a separate ReturnOrderLineItem representing that quantity.

API Version

$2.0

Requires Chatter

No

Signature

```java
public static ConnectApi.ReturnItemsOutputRepresentation returnItems(String returnOrderId, ConnectApi.ReturnItemsInputRepresentation returnItemsInput)
```

Parameters

`returnOrderId`
Type: `String`
ID of the ReturnOrder.

`returnItemsInput`
Type: `ConnectApi.ReturnItemsInputRepresentation`
Data about products and delivery charges to return, as well as associated return fees.
Return Value
Type: \texttt{ConnectApi.ReturnItemsOutputRepresentation}

SEE ALSO:
- createMultipleInvoices(invoicesInput)
- ensureRefundsAsync(orderSummaryId, ensureRefundsInput)
- createReturnOrder(returnOrderInput)

Routing Class
Route orders to inventory locations in Order Management.

Namespace
\texttt{ConnectApi}

Routing Methods
The following are methods for \texttt{Routing}. All methods are static.

\textbf{IN THIS SECTION:}
- confirmHeldFOCapacity(confirmHeldFOCapacityInput)
  Confirm held fulfillment order capacity at one or more locations. This call decreases a location's held capacity and increases its assigned fulfillment order count. Confirm held capacity when you assign a fulfillment order to a location.
- findRoutesWithFewestSplits(findRoutesWithFewestSplitsInputRepresentation)
  Returns combinations of inventory locations that can fulfill an order within a specified limit of shipment splits. By default, checks up to 1,000,000 potential routes, returning a maximum of 10,000 results.
- findRoutesWithFewestSplitsUsingOCI(findRoutesWithFewestSplitsUsingOCIInput)
  For one or more order summaries, find inventory availability using Omnichannel Inventory and identify the fulfillment routes with fewest splits. By default, checks up to 1,000,000 potential routes, returning a maximum of 10,000 results. This method combines the functionality of the getInventoryAvailability() and findRoutesWithFewestSplits() methods.
- getFOCapacityValues(getFOCapacityValuesInput)
  Get information about the current fulfillment order capacity of one or more locations.
- holdFOCapacity(holdFOCapacityInput)
  Hold fulfillment order capacity at a location. Holding capacity at a location reserves a space for a fulfillment order that you'll assign to it.
- rankAverageDistance(rankAverageDistanceInputRepresentation)
  Calculates the average distance from sets of inventory locations to an order recipient, and ranks them. Use this method to compare the average shipping distances for different sets of locations that can fulfill an order.
- releaseHeldFOCapacity(releaseHeldFOCapacityInput)
  Release held fulfillment order capacity at one or more locations. This call decreases a location's held capacity without changing its assigned fulfillment order count. Release held capacity when you cancel the assignment of a fulfillment order to a location.
confirmHeldFOCapacity(confirmHeldFOCapacityInput)

Confirm held fulfillment order capacity at one or more locations. This call decreases a location’s held capacity and increases its assigned fulfillment order count. Confirm held capacity when you assign a fulfillment order to a location.

API Version

55.0

Requires Chatter

No

Signature

public static ConnectApi.ConfirmHeldFOCapacityOutputRepresentation confirmHeldFOCapacity(ConnectApi.ConfirmHeldFOCapacityInputRepresentation confirmHeldFOCapacityInput)

Parameters

confirmHeldFOCapacityInput

Type: ConnectApi.ConfirmHeldFOCapacityInputRepresentation

The input includes, for each fulfillment order, the location where capacity is held for it.

Return Value

Type: ConnectApi.ConfirmHeldFOCapacityOutputRepresentation

findRoutesWithFewestSplits(findRoutesWithFewestSplitsInputRepresentation)

Returns combinations of inventory locations that can fulfill an order within a specified limit of shipment splits. By default, checks up to 1,000,000 potential routes, returning a maximum of 10,000 results.

API Version

51.0

Requires Chatter

No

Signature

public static ConnectApi.FindRoutesWithFewestSplitsOutputRepresentation findRoutesWithFewestSplits(ConnectApi.FindRoutesWithFewestSplitsInputRepresentation findRoutesWithFewestSplitsInputRepresentation)
Parameters

`findRoutesWithFewestSplitsInputRepresentation`
Type: `ConnectApi.FindRoutesWithFewestSplitsInputRepresentation`

The input includes the ordered item quantities, data about available inventory, and, optionally, a maximum allowable number of shipment splits.

Return Value
Type: `ConnectApi.FindRoutesWithFewestSplitsOutputRepresentation`

`findRoutesWithFewestSplitsUsingOCI(findRoutesWithFewestSplitsUsingOCIInput)`

For one or more order summaries, find inventory availability using Omnichannel Inventory and identify the fulfillment routes with fewest splits. By default, checks up to 1,000,000 potential routes, returning a maximum of 10,000 results. This method combines the functionality of the `getInventoryAvailability()` and `findRoutesWithFewestSplits()` methods.

API Version

54.0

Requires Chatter

No

Signature

`public static ConnectApi.FindRoutesWithFewestSplitsUsingOCIOutputRepresentation findRoutesWithFewestSplitsUsingOCI(ConnectApi.FindRoutesWithFewestSplitsUsingOCIInputRepresentation findRoutesWithFewestSplitsUsingOCIInput)`

Parameters

`findRoutesWithFewestSplitsUsingOCIInput`
Type: `ConnectApi.FindRoutesWithFewestSplitsUsingOCIInputRepresentation`

The input includes, for each order, the ordered item quantities, the assigned location group or locations, and, optionally, a maximum allowable number of shipment splits and a list of locations to exclude from the calculation.

Return Value
Type: `ConnectApi.FindRoutesWithFewestSplitsUsingOCIOutputRepresentation`

SEE ALSO:
- `getInventoryAvailability(inventoryAvailabilityInputRepresentation)`
- `findRoutesWithFewestSplits(findRoutesWithFewestSplitsInputRepresentation)`
- `getFOCapacityValues(getFOCapacityValuesInput)`

Get information about the current fulfillment order capacity of one or more locations.
API Version
55.0

Requires Chatter
No

Signature
public static ConnectApi.GetFOCapacityValuesOutputRepresentation
getFOCapacityValues(ConnectApi.GetFOCapacityValuesRequestInputRepresentation
getFOCapacityValuesInput)

Parameters
getFOCapacityValuesInput
Type: ConnectApi.GetFOCapacityValuesRequestInputRepresentation
Locations to get fulfillment order capacity information about.

Return Value
Type: ConnectApi.GetFOCapacityValuesOutputRepresentation

holdFOCapacity(holdFOCapacityInput)
Hold fulfillment order capacity at a location. Holding capacity at a location reserves a space for a fulfillment order that you’ll assign to it.

API Version
55.0

Requires Chatter
No

Signature
public static ConnectApi.HoldFOCapacityOutputRepresentation
holdFOCapacity(ConnectApi.HoldFOCapacityInputRepresentation holdFOCapacityInput)

Parameters
holdFOCapacityInput
Type: ConnectApi.HoldFOCapacityInputRepresentation
The input includes, for each fulfillment order, the location to hold capacity for it.

Return Value
Type: ConnectApi.HoldFOCapacityOutputRepresentation
rankAverageDistance(rankAverageDistanceInputRepresentation)
Calculates the average distance from sets of inventory locations to an order recipient, and ranks them. Use this method to compare the average shipping distances for different sets of locations that can fulfill an order.

API Version
51.0

Requires Chatter
No

Signature
public static ConnectApi.RankAverageDistanceOutputRepresentation rankAverageDistance(ConnectApi.RankAverageDistanceInputRepresentation rankAverageDistanceInputRepresentation)

Parameters
rankAverageDistanceInputRepresentation
Type: ConnectApi.RankAverageDistanceInputRepresentation
An order recipient’s geographic location and information about sets of inventory locations that can fulfill the order.

Return Value
Type: ConnectApi.RankAverageDistanceOutputRepresentation

releaseHeldFOCapacity(releaseHeldFOCapacityInput)
Release held fulfillment order capacity at one or more locations. This call decreases a location’s held capacity without changing its assigned fulfillment order count. Release held capacity when you cancel the assignment of a fulfillment order to a location.

API Version
55.0

Requires Chatter
No

Signature
public static ConnectApi.ReleaseHeldFOCapacityOutputRepresentation releaseHeldFOCapacity(ConnectApi.ReleaseHeldFOCapacityInputRepresentation releaseHeldFOCapacityInput)
Parameters

releaseHeldFOCapacityInput
  Type: ConnectApi.ReleaseHeldFOCapacityInputRepresentation
  The input includes, for each fulfillment order, the location that holds the capacity to release.

Return Value

Type: ConnectApi.ReleaseHeldFOCapacityOutputRepresentation

SalesforceInbox Class

Access information about Automated Activity Capture, which is available in Einstein and Salesforce Inbox.

Namespace

ConnectApi

SalesforceInbox Methods

The following are methods for SalesforceInbox. All methods are static.

IN THIS SECTION:

  shareActivity(activityId, sharingInfo)
  Share emails or events with certain groups of users.

shareActivity(activityId, sharingInfo)

Share emails or events with certain groups of users.

API Version

39.0

Requires Chatter

No

Signature

public static ConnectApi.ActivitySharingResult shareActivity(String activityId, ConnectApi.ActivitySharingInput sharingInfo)

Parameters

activityId
  Type: String
  The ID of the activity.
**sharingInfo**

Type: `ConnectApi.ActivitySharingInput`

A `ConnectApi.ActivitySharingInput` object.

**Return Value**

Type: `ConnectApi.ActivitySharingResult`

**Usage**

This method is a feature of both Sales Cloud Einstein and Inbox. It lets users connect their email and calendar to Salesforce. Then, their emails and events are automatically added to related Salesforce records. Users can specify who their individual emails and events are shared with.

**Sites Class**

Search an Experience Cloud site.

**Namespace**

`ConnectApi`

**Sites Methods**

The following are methods for `Sites`. All methods are static.

**IN THIS SECTION:**

- `searchSite(siteId, queryTerm, pageToken, pageSize, language)`
  Search an Experience Cloud site.

**searchSite**(siteId, queryTerm, pageToken, pageSize, language)

Search an Experience Cloud site.

**API Version**

54.0

**Available to Guest Users**

54.0

**Requires Chatter**

No
Signature

public static ConnectApi.SiteSearchResult searchSite(String siteId, String queryTerm, String pageToken, Integer pageSize, String language)

Parameters

siteId
Type: String
ID for the Experience Cloud site.

queryTerm
Type: String
White-space separated words used to search for relevant content. Provide a maximum of 1024 characters, composed of up to 32 words and white spaces. Logical operators aren’t supported.

pageToken
Type: String
Specifies the base64 encoded page token. Page tokens are returned as part of the response. If unspecified, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 240. If you pass in null, the default size is 25.

language
Type: String
Language locale for the context user. If unspecified or if the specified language locale isn’t available, the default value is en_US.

Return Value
Type: ConnectApi.SiteSearchResult

SmartDataDiscovery Class
Get predictions on Salesforce objects.
Use the ConnectApi.SmardDataDiscovery.predict method to get predictions on Salesforce objects. For more information, see Get Predictions in Apex.

SocialEngagement Class
Manage information about social accounts or fan pages for social networks.

Namespace
ConnectApi

SocialEngagement Methods
The following are methods for SocialEngagement. All methods are static.
IN THIS SECTION:

- `deleteSocialPost(socialPostId, socialAccountId)`
  Delete a social post from its social network.

- `followSocialPersona(socialPersonaId, socialAccountId)`
  Follow a social persona in its social network.

- `followSocialPostPersona(socialPostId, socialAccountId)`
  Follow a social persona on a social post in its social network.

- `getIntents(socialPostId)`
  Get available intents for a social post.

- `getManagedSocialAccount(id)`
  Get a managed social account that is in the org and assigned to the user.

- `getManagedSocialAccounts()`
  Gets a list of managed social accounts that are in the org and assigned to the user.

- `getManagedSocialAccounts(socialNetwork)`
  Get a list of managed social accounts that are in the org and assigned to the user.

- `getRelationship(id, socialPersonaId)`
  Get the follow relationship between a managed social account and a social persona.

- `hideSocialPost(socialPostId, socialAccountId)`
  Hide a social post in its social network.

- `likeSocialPost(socialPostId, socialAccountId)`
  Like a social post in its social network.

- `massApprove(massApproval)`
  Approve or reject the publishing of a large number of social posts.

- `recallApproval(socialPostId)`
  Recall an approval request to publish a social post.

- `unfollowSocialPersona(socialPersonaId, socialAccountId)`
  Stop following a social persona in its social network.

- `unfollowSocialPostPersona(socialPostId, socialAccountId)`
  Stop following a social persona of a social post in its social network.

- `unhideSocialPost(socialPostId, socialAccountId)`
  Unhide a social post in its social network.

- `unlikeSocialPost(socialPostId, socialAccountId)`
  Unlike a social post in its social network.

---

**deleteSocialPost(socialPostId, socialAccountId)**

Delete a social post from its social network.

**Note:** Deleting a social post from its social network doesn’t delete the record from Salesforce.
**API Version**

46.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.DeleteSocialPostIntent deleteSocialPost(String socialPostId, String socialAccountId)
```

**Parameters**

- **socialPostId**
  - Type: String
  - ID of the social post to delete.

- **socialAccountId**
  - Type: String
  - ID of the social account that deletes the post.

**Return Value**

Type: `ConnectApi.DeleteSocialPostIntent`

**followSocialPersona(socialPersonaId, socialAccountId)**

Follow a social persona in its social network.

**API Version**

45.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.FollowSocialPersonaIntent followSocialPersona(String socialPersonaId, String socialAccountId)
```

**Parameters**

- **socialPersonaId**
  - Type: String
  - ID of the social persona to follow.
socialAccountId

Type: **String**

ID of the social account that follows the social persona.

**Return Value**

Type: **ConnectApi.FollowSocialPersonaIntent**

```java
followSocialPostPersona(socialPostId, socialAccountId)
```

Follow a social persona on a social post in its social network.

**API Version**

45.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.FollowSocialPersonaIntent followSocialPostPersona(String socialPostId, String socialAccountId)
```

**Parameters**

`socailPostId`

Type: **String**

ID of the social post authored by the social persona to follow.

`socialAccountId`

Type: **String**

ID of the social account that follows the social persona.

**Return Value**

Type: **ConnectApi.FollowSocialPersonaIntent**

```java
getIntents(socialPostId)
```

Get available intents for a social post.

**API Version**

45.0

**Requires Chatter**

No
Signature
public static ConnectApi.SocialPostIntents getIntents(String socialPostId)

Parameters
socialPostId
Type: String
ID of a social post.

Return Value
Type: ConnectApi.SocialPostIntents

getManagedSocialAccount(id)
Get a managed social account that is in the org and assigned to the user.

API Version
44.0

Requires Chatter
No

Signature
public static ConnectApi.ManagedSocialAccount getManagedSocialAccount(String id)

Parameters
id
Type: String
Description: Internal SFDC ID for this managed social account.

Return Value
Type: ConnectApi.ManagedSocialAccount

getManagedSocialAccounts()
Gets a list of managed social accounts that are in the org and assigned to the user.

API Version
44.0

Requires Chatter
No
**Signature**

```java
public static ConnectApi.ManagedSocialAccounts getManagedSocialAccounts()
```

**Return Value**

Type: `ConnectApi.ManagedSocialAccounts`

**getManagedSocialAccounts(socialNetwork)**

Get a list of managed social accounts that are in the org and assigned to the user.

**API Version**

44.0

**Requires Chatter**

No

**Signature**

```java
public static ConnectApi.ManagedSocialAccounts getManagedSocialAccounts(ConnectApi.SocialNetworkProvider socialNetwork)
```

**Parameters**

`socialNetwork`

Type: `ConnectApi.SocialNetworkProvider`

Description: Filters results based on the social network. Values are:

- Facebook
- GooglePlus
- Instagram
- InstagramBusiness
- KakaoTalk
- Kik
- Line
- LinkedIn
- Messenger
- Other
- Pinterest
- QQ
- Rypple
- SinaWeibo
- SMS
- Snapchat
Return Value
Type: ConnectApi.ManagedSocialAccounts

getRelationship(id, socialPersonaId)
Get the follow relationship between a managed social account and a social persona.

API Version
46.0

Requires Chatter
No

Signature
public static ConnectApi.SocialAccountRelationship getRelationship(String id, String socialPersonaId)

Parameters
id
Type: String
ID of the managed social account.

socialPersonaId
Type: String
ID of the social persona.

Return Value
Type: ConnectApi.SocialAccountRelationship

hideSocialPost(socialPostId, socialAccountId)
Hide a social post in its social network.

API Version
46.0
Requires Chatter
No

Signature

public static ConnectApi.HideSocialPostIntent hideSocialPost(String socialPostId, String socialAccountId)

Parameters

socialPostId
Type: String
ID of the social post to hide.
socialAccountId
Type: String
ID of the social account that hides the post.

Return Value
Type: ConnectApi.HideSocialPostIntent

likeSocialPost(socialPostId, socialAccountId)
Like a social post in its social network.

API Version
46.0

Requires Chatter
No

Signature

public static ConnectApi.LikeSocialPostIntent likeSocialPost(String socialPostId, String socialAccountId)

Parameters

socialPostId
Type: String
ID of the social post to like.
socialAccountId
Type: String
ID of the social account that likes the post.
Return Value
Type: `ConnectApi.LikeSocialPostIntent`

```java
massApprove(massApproval)
```
Approve or reject the publishing of a large number of social posts.

**API Version**
46.0

**Requires Chatter**
No

**Signature**
```java
public static ConnectApi.SocialPostMassApprovalOutput
massApprove(ConnectApi.SocialPostMassApprovalInput massApproval)
```

**Parameters**
- `massApproval`
  Type: `ConnectApi.SocialPostMassApprovalInput`
  A `ConnectApi.SocialPostMassApprovalInput` body that includes a list of social post IDs and the action to approve or reject publishing them.

**Return Value**
Type: `ConnectApi.SocialPostMassApprovalOutput`

```java
recallApproval(socialPostId)
```
Recall an approval request to publish a social post.

**API Version**
46.0

**Requires Chatter**
No

**Signature**
```java
public static Void recallApproval(String socialPostId)
```
Parameters

socialPostId
Type: String
ID of the social post.

Return Value
Type: Void

unfollowSocialPersona(socialPersonaId, socialAccountId)
Stop following a social persona in its social network.

API Version
45.0

Requires Chatter
No

Signature
public static Void unfollowSocialPersona(String socialPersonaId, String socialAccountId)

Parameters

socialPersonaId
Type: String
ID of the social persona to stop following.

socialAccountId
Type: String
ID of the social account that stops following the social persona.

Return Value
Type: Void

unfollowSocialPostPersona(socialPostId, socialAccountId)
Stop following a social persona of a social post in its social network.

API Version
45.0

Requires Chatter
No
**unfollowSocialPostPersona**

**Signature**

```java
public static Void unfollowSocialPostPersona(String socialPostId, String socialAccountId)
```

**Parameters**

- `socialPostId`
  - Type: `String`
  - ID of the social post authored by the social persona to stop following.
- `socialAccountId`
  - Type: `String`
  - ID of the social account that stops following the social persona.

**Return Value**

Type: Void

**unhideSocialPost**

**Signature**

```java
public static Void unhideSocialPost(String socialPostId, String socialAccountId)
```

**Parameters**

- `socialPostId`
  - Type: `String`
  - ID of the social post to unhide.
- `socialAccountId`
  - Type: `String`
  - ID of the social account that unhides the post.

**Return Value**

Type: Void

**unlikeSocialPost**

**Signature**

```java
public static Void unlikeSocialPost(String socialPostId, String socialAccountId)
```

**Parameters**

- `socialPostId`
  - Type: `String`
  - ID of the social post in its social network.

**API Version**

46.0

**Requires Chatter**

No
API Version
46.0

Requires Chatter
No

Signature
public static Void unlikeSocialPost(String socialPostId, String socialAccountId)

Parameters
socialPostId
Type: String
ID of the social post to unlike.
socialAccountId
Type: String
ID of the social account that unlikes the post.

Return Value
Type: Void

Surveys Class
Send survey invitations by email.

Namespace
ConnectApi

Surveys Methods
These methods are for Surveys. All methods are static.

IN THIS SECTION:

sendSurveyInvitationEmail(surveyID, SurveyEmailInput)
Email survey invitations to up to 300 participants. You can email either leads, contacts, or users in your org. Either a link to launch the survey or a question can be embedded in the email invitations.

sendSurveyInvitationEmail(surveyID, SurveyEmailInput)
Email survey invitations to up to 300 participants. You can email either leads, contacts, or users in your org. Either a link to launch the survey or a question can be embedded in the email invitations.
API Version
50.0

Requires Chatter
No

Signature
public static ConnectApi.SurveyInvitationEmailOutput sendSurveyInvitationEmail(String surveyID, ConnectApi.SurveyInvitationEmailInput SurveyEmailInput)

Parameters
surveym
  Type: String
  ID of the survey.
SurveyEmailInput
  Type: ConnectApi.SurveyInvitationEmailInput
  A ConnectApi.SurveyInvitationEmailInput object.

Return Value
Type: ConnectApi.SurveyInvitationEmailOutput

TaxPlatform Class
Apply or cancel tax.

Namespace
ConnectApi

TaxPlatform Methods
The following are methods for TaxPlatform. All methods are static.

IN THIS SECTION:
calculateTax(calculateTax)
  Apply tax or cancel tax.

calculateTax(calculateTax)
  Apply tax or cancel tax.
API Version
55.0

Requires Chatter
No

Signature
global static ConnectApi.CalculateTaxResponse calculateTax(ConnectApi.CalculateTaxRequest calculateTax)

Parameters
calculateTax
Type: ConnectApi.CalculateTaxRequest
Represents a request to calculate tax for one or more line items.

Return Value
Type: ConnectApi.CalculateTaxResponse

Topics Class
Access information about topics, such as their descriptions, the number of people talking about them, related topics, and information about groups contributing to the topic. Update a topic’s name or description, merge topics, and add and remove topics from records and feed items.

Namespace
ConnectApi

Topics Methods
The following are methods for Topics. All methods are static.

IN THIS SECTION:
assignTopic(communityld, recordld, topicId)
Assign a topic to a record or feed item.
assignTopicByName(communityld, recordld, topicName)
Assign a topic to a record or feed item.
createTopic(communityld, name, description)
Create a topic.
createTopicDataCategoryRules(communityld, dataCategoryGroup, dataCategory, topicNames)
Create topic and article assignment rules by data category.
deleteTopic(communityId, topicId)
Delete a topic.

getGroupsRecentlyTalkingAboutTopic(communityId, topicId)
Get information about the five groups that most recently contributed to a topic.

getRecentlyTalkingAboutTopicsForGroup(communityId, groupId)
Get up to five topics most recently used in a group.

getRecentlyTalkingAboutTopicsForUser(communityId, userId)
Get up to five topics most recently used by a user.

getRelatedTopics(communityId, topicId)
Get up to five topics most closely related to a topic.

getTopic(communityId, topicId)
Get a topic.

getTopicDataCategoryRules(communityId, dataCategoryGroup, dataCategory)
Get topic and article assignment rules by data category.

getTopics(communityId, recordId)
Get the first page of topics assigned to a record or feed item.

getTopics(communityId)
Get the first page of topics for the org or Experience Cloud site.

getTopics(communityId, sortParam)
Get the first page of sorted topics for the org or community.

getTopics(communityId, pageParam, pageSize)
Get a page of topics.

getTopics(communityId, pageParam, pageSize, sortParam)
Get a page of sorted topics.

getTopics(communityId, q, sortParam)
Get the sorted topics that match the search criteria.

getTopics(communityId, q, pageParam, pageSize)
Get a page of topics that match the search criteria.

getTopics(communityId, q, pageParam, pageSize, sortParam)
Get a page of sorted topics that match the search criteria.

getTopics(communityId, q, exactMatch)
Get the topic that matches the exact, case-insensitive name.

getTopicsOrFallbackToRenamedTopics(communityId, q, exactMatch, fallBackToRenamedTopics)
Get the most recent renamed topic match, if there isn't an exact match.

getTopicSuggestions(communityId, recordId, maxResults)
Get up to a specified number of suggested topics for a record or feed item.

getTopicSuggestions(communityId, recordId)
Get suggested topics for a record or feed item.

getTopicSuggestionsForText(communityId, text, maxResults)
Get up to a specified number of suggested topics for a string of text.
getTopicSuggestionsForText(communityId, text)
Get suggested topics for a string of text.

getTrendingTopics(communityId)
Get trending topics for the org or Experience Cloud site.

getTrendingTopics(communityId, maxResults)
Get up to a specified number of trending topics for the org or Experience Cloud site.

mergeTopics(communityId, topicId, idsToMerge)
Merge up to five secondary topics with a primary topic.

reassignTopicDataCategoryRules(communityId, dataCategoryGroup, dataCategory, topicNames)
Reassign topic and article assignment rules by data category by deleting the existing rules and creating new rules.

reassignTopicsByName(communityId, recordId, topicNames)
Reassign all the topics on a record or feed item, that is, remove all the assigned topics on a record or feed item and add topics. Optionally, provide a list of suggested topics to assign to a record or feed item to improve future topic suggestions.

unassignTopic(communityId, recordId, topicId)
Remove a topic from a record or feed item.

updateTopic(communityId, topicId, topic)
Update the description or name of a topic or merge up to five secondary topics with a primary topic.

updateTopicsForArticlesInDataCategory(communityId, dataCategoryGroup, dataCategory, articleTopicAssignmentJob)
Assign topics to articles and unassign topics from articles in a data category.

### assignTopic(communityId, recordId, topicId)
Assign a topic to a record or feed item.

#### API Version
29.0

#### Requires Chatter
No

#### Signature
```java
public static ConnectApi.Topic assignTopic(String communityId, String recordId, String topicId)
```

#### Parameters
- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.
- **recordId**
  - Type: `String`
  - ID for a record or feed item.
**topicId**  
Type: **String**  
ID for a topic.

**Return Value**  
Type: **ConnectApi.Topic**

**Usage**  
Only users with the Assign Topics permission can add existing topics to records or feed items. Administrators must enable topics for objects before users can add topics to records of that object type.

**assignTopicByName** (communityId, recordId, topicName)

Assign a topic to a record or feed item.

**API Version**  
29.0

**Requires Chatter**  
No

**Signature**

```java
public static ConnectApi.Topic assignTopicByName(String communityId, String recordId, String topicName)
```

**Parameters**

- **communityId**  
  Type: **String**  
  ID for an Experience Cloud site, internal, or null.

- **recordId**  
  Type: **String**  
  The ID of the record or feed item to which to assign the topic.

- **topicName**  
  Type: **String**  
  The name of a new or existing topic.

**Return Value**  
Type: **ConnectApi.Topic**
Usage
Only users with the Assign Topics permission can add existing topics to records or feed items. Only users with the Create Topics permission can add new topics to records or feed items. Administrators must enable topics for objects before users can add topics to records of that object type.

`createTopic(communityId, name, description)`
Create a topic.

API Version
36.0

Requires Chatter
No

Signature
`public static ConnectApi.Topic createTopic(String communityId, String name, String description)`

Parameters
- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, internal, or null.
- **name**
  - Type: `String`
  - The name of the topic.
- **description**
  - Type: `String`
  - The description of the topic.

Return Value
Type: `ConnectApi.Topic`

Usage
Only users with the Create Topics permission can create a topic.

`createTopicDataCategoryRules(communityId, dataCategoryGroup, dataCategory, topicNames)`
Create topic and article assignment rules by data category.
API Version
40.0

Requires Chatter
No

Signature
public static ConnectApi.TopicPage createTopicDataCategoryRules(String communityId,
String dataCategoryGroup, String dataCategory, ConnectApi.TopicNamesInput topicNames)

Parameters

communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

dataCategoryGroup
  Type: String
  The data category group used by articles.

dataCategory
  Type: String
  The data category used by articles.

topicNames
  Type: ConnectApi.TopicNamesInput
  A ConnectApi.TopicNamesInput object with the names of topics to assign to articles in a data category.

Return Value
Type: ConnectApi.TopicPage

deleteTopic(communityId, topicId)
Delete a topic.

API Version
29.0

Requires Chatter
No

Signature
public static Void deleteTopic(String communityId, String topicId)
Parameters

communityId
Type: String,
ID for an Experience Cloud site, internal, or null.

topicId
Type: String
ID for a topic.

Return Value
Type: Void

Usage
Only users with the Delete Topics or Modify All Data permission can delete topics.

Topic deletion is asynchronous. If a topic is requested before the deletion completes, the response is successful and the isBeingDeleted property of ConnectApi.Topic is true in version 33.0 and later. If a topic is requested after the deletion completes, the response is ConnectApi.NotFoundException.

groupsRecentlyTalkingAboutTopic(communityId, topicId)
Get information about the five groups that most recently contributed to a topic.

API Version
29.0

Available to Guest Users
32.0

Requires Chatter
Yes

Signature
public static ConnectApi.ChatterGroupSummaryPage getGroupsRecentlyTalkingAboutTopic(String communityId, String topicId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

topicId
Type: String
ID for a topic.
Return Value

Type: ConnectApi.ChatterGroupSummaryPage

Usage

To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- setTestGetGroupsRecentlyTalkingAboutTopic(communityId, topicId, result)

getRecentlyTalkingAboutTopicsForGroup(communityId, groupId)

Get up to five topics most recently used in a group.

API Version

29.0

Available to Guest Users

32.0

Requires Chatter

Yes

Signature

public static ConnectApi.TopicPage getRecentlyTalkingAboutTopicsForGroup(String communityId, String groupId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

groupId
Type: String
ID for a group.

Return Value

Type: ConnectApi.TopicPage
Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestGetRecentlyTalkingAboutTopicsForGroup(communityId, groupId, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

getRecentlyTalkingAboutTopicsForUser(communityId, userId)
Get up to five topics most recently used by a user.

API Version
29.0

Available to Guest Users
32.0

Requires Chatter
Yes

Signature
```
public static ConnectApi.TopicPage getRecentlyTalkingAboutTopicsForUser(String communityId, String userId)
```

Parameters
- `communityId`
  Type: String
  ID for an Experience Cloud site, internal, or `null`.
- `userId`
  Type: String
  ID for a user.

Return Value
Type: `ConnectApi.TopicPage`
Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestGetRecentlyTalkingAboutTopicsForUser(communityId, userId, result)`
  - *Apex Developer Guide: Testing ConnectApi Code*

**getRelatedTopics(communityId, topicId)**
Get up to five topics most closely related to a topic.
Two topics that are assigned to the same feed item at least three times are related.

**API Version**
29.0

**Available to Guest Users**
32.0

**Requires Chatter**
No

**Signature**
```java
public static ConnectApi.TopicPage getRelatedTopics(String communityId, String topicId)
```

**Parameters**
- `communityId`
  - Type: String
  - ID for an Experience Cloud site, internal, or null.
- `topicId`
  - Type: String
  - ID for a topic.

**Return Value**
Type: `ConnectApi.TopicPage`
Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestGetRelatedTopics(communityId, topicId, result)

getTopic(communityId, topicId)
Get a topic.

API Version
29.0

Available to Guest Users
32.0

Requires Chatter
No

Signature
public static ConnectApi.Topic getTopic(String communityId, String topicId)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

topicId
Type: String
ID for a topic.

Return Value
Type: ConnectApi.Topic

getTopicDataCategoryRules(communityId, dataCategoryGroup, dataCategory)
Get topic and article assignment rules by data category.

API Version
40.0
Requires Chatter
No

Signature
public static ConnectApi.TopicPage getTopicDataCategoryRules(String communityId, String dataCategoryGroup, String dataCategory)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

dataCategoryGroup
Type: String
The data category group used by articles.

dataCategory
Type: String
The data category used by articles.

Return Value
Type: ConnectApi.TopicPage

getTopics(communityId, recordId)
Get the first page of topics assigned to a record or feed item.

API Version
29.0

Available to Guest Users
32.0

Requires Chatter
No

Signature
public static ConnectApi.TopicPage getTopics(String communityId, String recordId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

recordId
   Type: String
   ID for a record or feed item.

Return Value
Type: ConnectApi.TopicPage

Usage
Administrators must enable topics for objects before users can add topics to records of that object type.

getTopics(communityId)
Get the first page of topics for the org or Experience Cloud site.

API Version
29.0

Available to Guest Users
32.0

Requires Chatter
No

Signature
public static ConnectApi.TopicPage getTopics(String communityId)

Parameters
communityId
   Type: String
   ID for an Experience Cloud site, internal, or null.

Return Value
Type: ConnectApi.TopicPage

getTopics(communityId, sortParam)
Get the first page of sorted topics for the org or community.

API Version
29.0
Available to Guest Users
32.0

Requires Chatter
No

Signature
public static ConnectApi.TopicPage getTopics(String communityId, ConnectApi.TopicSort sortParam)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

sortParam
Type: ConnectApi.TopicSort
Values are:
- popularDesc—Sorts topics by popularity with the most popular first. This value is the default.
- alphaAsc—Sorts topics alphabetically.

Return Value
Type: ConnectApi.TopicPage

getTopics(communityId, pageParam, pageSize)
Get a page of topics.

API Version
29.0

Available to Guest Users
32.0

Requires Chatter
No

Signature
public static ConnectApi.TopicPage getTopics(String communityId, Integer pageParam, Integer pageSize)
Parameters

* **communityId**
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.

* **pageParam**
  Type: `Integer`
  Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

* **pageSize**
  Type: `Integer`
  Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

Return Value

Type: `ConnectApi.TopicPage`

`getTopics(communityId, pageParam, pageSize, sortParam)`

Get a page of sorted topics.

API Version

29.0

Available to Guest Users

32.0

Requires Chatter

No

Signature

`public static ConnectApi.TopicPage getTopics(String communityId, Integer pageParam, Integer pageSize, ConnectApi.TopicSort sortParam)`

Parameters

* **communityId**
  Type: `String`
  ID for an Experience Cloud site, internal, or `null`.

* **pageParam**
  Type: `Integer`
  Number of the page you want returned. Starts at 0. If you pass in `null` or 0, the first page is returned.

* **pageSize**
  Type: `Integer`
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

`sortParam`
- Type: `ConnectApi.TopicSort`
- Values are:
  - `popularDesc`—Sorts topics by popularity with the most popular first. This value is the default.
  - `alphaAsc`—Sorts topics alphabetically.

Return Value
- Type: `ConnectApi.TopicPage`

`getTopics(communityId, q, sortParam)`
Get the sorted topics that match the search criteria.

API Version
- 29.0

Available to Guest Users
- 32.0

Requires Chatter
- No

Signature
```
public static ConnectApi.TopicPage getTopics(String communityId, String q, ConnectApi.TopicSort sortParam)
```

Parameters
- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, `internal`, or `null`.

- `q`
  - Type: `String`
  - Specifies the string to search. The string must contain at least two characters, not including wildcards.

- `sortParam`
  - Type: `ConnectApi.TopicSort`
  - Values are:
    - `popularDesc`—Sorts topics by popularity with the most popular first. This value is the default.
    - `alphaAsc`—Sorts topics alphabetically.
Return Value
Type: ConnectApi.TopicPage

getTopics(communityId, q, pageParam, pageSize)
Get a page of topics that match the search criteria.

API Version
29.0

Available to Guest Users
32.0

Requires Chatter
No

Signature
public static ConnectApi.TopicPage getTopics(String communityId, String q, Integer pageParam, Integer pageSize)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
Specifies the string to search. The string must contain at least two characters, not including wildcards.

pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

Return Value
Type: ConnectApi.TopicPage

getTopics(communityId, q, pageParam, pageSize, sortParam)
Get a page of sorted topics that match the search criteria.
API Version
29.0

Available to Guest Users
32.0

Requires Chatter
No

Signature
public static ConnectApi.TopicPage getTopics(String communityId, String q, Integer pageParam, Integer pageSize, ConnectApi.TopicSort sortParam)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
Specifies the string to search. The string must contain at least two characters, not including wildcards.

pageParam
Type: Integer
Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

sortParam
Type: ConnectApi.TopicSort
Values are:
• popularDesc—Sorts topics by popularity with the most popular first. This value is the default.
• alphaAsc—Sorts topics alphabetically.

Return Value
Type: ConnectApi.TopicPage

g getTopics(communityId, q, exactMatch)
Get the topic that matches the exact, case-insensitive name.
**API Version**
33.0

**Available to Guest Users**
33.0

**Requires Chatter**
No

**Signature**
```
public static ConnectApi.TopicPage getTopics(String communityId, String q, Boolean exactMatch)
```

**Parameters**
- **communityId**
  - Type: `String`
  - ID for an Experience Cloud site, `internal`, or `null`.
- **q**
  - Type: `String`
  - Specifies the string to search. The string must contain at least two characters, not including wildcards.
- **exactMatch**
  - Type: `Boolean`
  - Specify `true` to find a topic by its exact, case-insensitive name.

**Return Value**
Type: `ConnectApi.TopicPage`

**getTopicsOrFallBackToRenamedTopics(communityId, q, exactMatch, fallBackToRenamedTopics)**
Get the most recent renamed topic match, if there isn’t an exact match.

**API Version**
35.0

**Available to Guest Users**
35.0

**Requires Chatter**
No
public static ConnectApi.TopicPage getTopicsOrFallBackToRenamedTopics(String communityId, String q, Boolean exactMatch, Boolean fallBackToRenamedTopics)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

q
Type: String
Specifies the string to search. The string must contain at least two characters, not including wildcards.

exactMatch
Type: Boolean
Specify true to find a topic by its exact, case-insensitive name or to find the most recent renamed topic match if there isn’t an exact match.

fallBackToRenamedTopics
Type: Boolean
Specify true and if there isn’t an exact match, the most recent renamed topic match is returned. If there are multiple renamed topic matches, only the most recent is returned. If there are no renamed topic matches, an empty collection is returned.

Return Value
Type: ConnectApi.TopicPage

getTopicSuggestions(communityId, recordId, maxResults)
Get up to a specified number of suggested topics for a record or feed item.

API Version
29.0

Requires Chatter
No

public static ConnectApi.TopicSuggestionPage getTopicSuggestions(String communityId, String recordId, Integer maxResults)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.
recordId
  Type: String
  ID for a record or feed item.
maxResults
  Type: Integer
  Maximum number of topic suggestions that get returned. The default is 5. Value must be greater than 0 and less than or equal to 25.

Return Value
Type: ConnectApi.TopicSuggestionPage

Usage
Administrators must enable topics for objects before users can see suggested topics for records of that object type.
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
  setTestGetTopicSuggestions(communityId, recordId, maxResults, result)

getTopicSuggestions(communityId, recordId)
Get suggested topics for a record or feed item.

API Version
29.0

Requires Chatter
No

Signature
public static ConnectApi.TopicSuggestionPage getTopicSuggestions(String communityId, String recordId)

Parameters
  communityId
    Type: String
    ID for an Experience Cloud site, internal, or null.
  recordId
    Type: String
    ID for a record or feed item.
Return Value
Type: ConnectApi.TopicSuggestionPage

Usage
Administrators must enable topics for objects before users can see suggested topics for records of that object type.
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestGetTopicSuggestions(communityId, recordId, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

getTopicSuggestionsForText(communityId, text, maxResults)
Get up to a specified number of suggested topics for a string of text.

API Version
29.0

Requires Chatter
No

Signature
public static ConnectApi.TopicSuggestionPage getTopicSuggestionsForText(String communityId, String text, Integer maxResults)

Parameters
- `communityId`
  Type: String
  ID for an Experience Cloud site, internal, or null.
- `text`
  Type: String
  String of text.
- `maxResults`
  Type: Integer
  Maximum number of topic suggestions that get returned. The default is 5. Value must be greater than 0 and less than or equal to 25.

Return Value
Type: ConnectApi.TopicSuggestionPage
Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestGetTopicSuggestionsForText(communityId, text, maxResults, result)`
  *Apex Developer Guide: Testing ConnectApi Code*

`getTopicSuggestionsForText(communityId, text)`
Get suggested topics for a string of text.

API Version
29.0

Requires Chatter
No

Signature
```java
public static ConnectApi.TopicSuggestionPage getTopicSuggestionsForText(String communityId, String text)
```

Parameters
- `communityId`
  *Type: String*
  
  ID for an Experience Cloud site, `internal`, or `null`.

- `text`
  *Type: String*
  
  String of text.

Return Value
- Type: `ConnectApi.TopicSuggestionPage`

Usage
To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
- `setTestGetTopicSuggestionsForText(communityId, text, result)`
  *Apex Developer Guide: Testing ConnectApi Code*
**getTrendingTopics(communityId)**
Get trending topics for the org or Experience Cloud site.

**API Version**
29.0

**Available to Guest Users**
32.0

**Requires Chatter**
No

**Signature**
```
public static ConnectApi.TopicPage getTrendingTopics(String communityId)
```

**Parameters**

- `communityId`
  
  Type: `String`
  
  ID for an Experience Cloud site, `internal`, or `null`.

**Return Value**
Type: `ConnectApi.TopicPage`

**Usage**
The more frequently people add a specific topic to their posts and comments and comment on or like posts with the same topic over a short period, the more likely it is to become a trending topic. For example, if your coworkers are attending the upcoming Dreamforce conference and have started discussing it in Chatter, you might see a trending topic for Dreamforce. A trending topic is not solely based on popularity and usually relates to a one-time or infrequent event that has a spike in activity, such as a conference or a project deadline.

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

**SEE ALSO:**
- `setTestGetTrendingTopics(communityId, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

**getTrendingTopics(communityId, maxResults)**
Get up to a specified number of trending topics for the org or Experience Cloud site.

**API Version**
29.0
Available to Guest Users
32.0

Requires Chatter
No

Signature

public static ConnectApi.TopicPage getTrendingTopics(String communityId, Integer maxResults)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

maxResults
Type: Integer
Maximum number of topic suggestions that get returned. The default is 5. Value must be greater than 0 and less than or equal to 25.

Return Value
Type: ConnectApi.TopicPage

Usage

The more frequently people add a specific topic to their posts and comments and comment on or like posts with the same topic over a short period, the more likely it is to become a trending topic. For example, if your coworkers are attending the upcoming Dreamforce conference and have started discussing it in Chatter, you might see a trending topic for Dreamforce. A trending topic is not solely based on popularity and usually relates to a one-time or infrequent event that has a spike in activity, such as a conference or a project deadline.

To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

setTestGetTrendingTopics(communityId, maxResults, result)

mergeTopics(communityId, topicId, idsToMerge)

Merge up to five secondary topics with a primary topic.

API Version
33.0
Requires Chatter
No

Signature
```
public static ConnectApi.Topic mergeTopics(String communityId, String topicId,
List<String> idsToMerge)
```

Parameters
```
communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

topicId
  Type: String
  The ID for the primary topic for the merge. If this topic is a managed topic, it retains its topic type, topic images, and children topics.

idsToMerge
  Type: List<String>
  A list of up to five comma-separated secondary topic IDs to merge with the primary topic. If any of the secondary topics are navigational or featured topics, they lose their topic type, topic images, and children topics. Their feed items are reassigned to the primary topic. If you merge a topic with a content topic, the content associations are preserved. If you merge a topic with an inactive endorsee, the endorsement isn’t mapped to the primary topic.
```

Return Value
```
Type: ConnectApi.Topic
```

Usage
Only users with the Delete Topics or Modify All Data permission can merge topics.

```
reassignTopicDataCategoryRules(communityId, dataCategoryGroup, dataCategory,
topicNames)
```

Reassign topic and article assignment rules by data category by deleting the existing rules and creating new rules.

API Version
40.0

Requires Chatter
No

Signature
```
public static ConnectApi.TopicPage reassignTopicDataCategoryRules(String communityId,
String dataCategoryGroup, String dataCategory, ConnectApi.TopicNamesInput topicNames)
```
Parameters

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- **dataCategoryGroup**
  - Type: String
  - The data category group used by articles.

- **dataCategory**
  - Type: String
  - The data category used by articles.

- **topicNames**
  - Type: ConnectApi.TopicNamesInput
  - A ConnectApi.TopicNamesInput object with the names of topics to reassign to articles in a data category.

Return Value

Type: ConnectApi.TopicPage

**reassignTopicsByName(communityId, recordId, topicNames)**

Reassign all the topics on a record or feed item, that is, remove all the assigned topics on a record or feed item and add topics. Optionally, provide a list of suggested topics to assign to a record or feed item to improve future topic suggestions.

API Version

35.0

Requires Chatter

No

Signature

public static ConnectApi.TopicPage reassignTopicsByName(String communityId, String recordId, ConnectApi.TopicNamesInput topicNames)

Parameters

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or null.

- **recordId**
  - Type: String
  - The ID of the record or feed item to which to assign the topic.

- **topicNames**
  - Type: ConnectApi.TopicNamesInput
A list of topics to replace the currently assigned topics. Optionally, a list of suggested topics to assign to improve future topic suggestions.

**Return Value**
Type: `ConnectApi.TopicPage`

**Usage**
Only users with the Assign Topics permission can remove topics from records or feed items and add existing topics to records or feed items. Only users with the Create Topics permission can add new topics to records or feed items. Administrators must enable topics for objects before users can add topics to records of that object type.

### `unassignTopic(communityId, recordId, topicId)`
Remove a topic from a record or feed item.

**API Version**
29.0

**Requires Chatter**
No

**Signature**

```java
public static Void unassignTopic(String communityId, String recordId, String topicId)
```

**Parameters**

- `communityId`
  Type: `String`
  ID for an Experience Cloud site, internal, or null.

- `recordId`
  Type: `String`
  ID for a record or feed item.

- `topicId`
  Type: `String`
  ID for a topic.

**Return Value**
Type: Void

**Usage**
Only users with the Assign Topics permission can remove topics from feed items or records. Administrators must enable topics for objects before users can add topics to records of that object type.
updateTopic(communityId, topicId, topic)
Update the description or name of a topic or merge up to five secondary topics with a primary topic.

API Version
29.0

Requires Chatter
No

Signature
public static ConnectApi.Topic updateTopic(String communityId, String topicId, ConnectApi.TopicInput topic)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

topicId
Type: String
ID for a topic.

topic
Type: ConnectApi.TopicInput
A ConnectApi.TopicInput object containing the name and description of the topic or up to five comma-separated secondary topic IDs to merge with the primary topic.

Return Value
Type: ConnectApi.Topic

Usage
Only users with the Edit Topics or Modify All Data permission can update topic names and descriptions. Only users with the Delete Topics or Modify All Data permission can merge topics.

updateTopicsForArticlesInDataCategory(communityId, dataCategoryGroup, dataCategory, articleTopicAssignmentJob)
Assign topics to articles and unassign topics from articles in a data category.

API Version
40.0
Requires Chatter
No

Signature
public static ConnectApi.TopicPage updateTopicsForArticlesInDataCategory(String
communityId, String dataCategoryGroup, String dataCategory,
ConnectApi.ArticleTopicAssignmentJobInput articleTopicAssignmentJob)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.
dataCategoryGroup
Type: String
The data category group used by articles.
dataCategory
Type: String
The data category used by articles.
articleTopicAssignmentJob
Type: ConnectApi.ArticleTopicAssignmentJobInput
A ConnectApi.ArticleTopicAssignmentJobInput object that indicates the operation to take on which topics.

Return Value
Type: ConnectApi.TopicPage

Topics Test Methods
The following are the test methods for Topics. All methods are static.
For information about using these methods to test your ConnectApi code, see Testing ConnectApi Code.

setTestGetGroupsRecentlyTalkingAboutTopic(communityId, topicId, result)
Register a ConnectApi.ChatterGroupSummaryPage object to be returned when the matching
ConnectApi.getGroupsRecentlyTalkingAboutTopic method is called in a test context. Use the method with the
same parameters or you receive an exception.

API Version
29.0

Signature
public static Void setTestGetGroupsRecentlyTalkingAboutTopic(String communityId, String
topicId, ConnectApi.ChatterGroupSummaryPage result)
Parameters

communityId  
Type: String
ID for an Experience Cloud site, internal, or null.

topicId  
Type: String
ID for a topic.

result  
Type: ConnectApi.ChatterGroupSummaryPage
Object containing test data.

Return Value

Type: Void

SEE ALSO:
getGroupsRecentlyTalkingAboutTopic(communityId, topicId)  

setTestGetRecentlyTalkingAboutTopicsForGroup(communityId, groupId, result)
Register a ConnectApi.TopicPage object to be returned when the matching
ConnectApi.getRecentlyTalkingAboutTopicsForGroup method is called in a test context. Use the method with
the same parameters or you receive an exception.

API Version

29.0

Signature

public static Void setTestGetRecentlyTalkingAboutTopicsForGroup(String communityId, 
String groupId, ConnectApi.TopicPage result)

Parameters

communityId  
Type: String
ID for an Experience Cloud site, internal, or null.

groupId  
Type: String
ID for a group.

result  
Type: ConnectApi.TopicPage
Object containing test data.
Return Value
Type: Void

SEE ALSO:
getRecentlyTalkingAboutTopicsForGroup(communityId, groupId)

**setTestGetRecentlyTalkingAboutTopicsForUser(communityId, userId, result)**
Register a ConnectApi.TopicPage object to be returned when the matching ConnectApi.getRecentlyTalkingAboutTopicsForUser method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
29.0

Signature
public static Void setTestGetRecentlyTalkingAboutTopicsForUser(String communityId, String userId, ConnectApi.TopicPage result)

Parameters
- **communityId**
  Type: String
  ID for an Experience Cloud site, internal, or null.
- **userId**
  Type: String
  ID for a user.
- **result**
  Type: ConnectApi.TopicPage
  Specify the test topics page.

Return Value
Type: Void

SEE ALSO:
getRecentlyTalkingAboutTopicsForUser(communityId, userId)

**setTestGetRelatedTopics(communityId, topicId, result)**
Register a ConnectApi.TopicPage object to be returned when the matching ConnectApi.getRelatedTopics method is called in a test context. Use the method with the same parameters or you receive an exception.
**API Version**

29.0

**Signature**

```java
public static Void setTestGetRelatedTopics(String communityId, String topicId,
ConnectApi.TopicPage result)
```

**Parameters**

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- `topicId`
  - Type: `String`
  - ID for a topic.

- `result`
  - Type: `ConnectApi.TopicPage`
  - Object containing test data.

**Return Value**

Type: `Void`

**SEE ALSO:**

- `getRelatedTopics(communityId, topicId)`

---

**setTestGetTopicSuggestions(communityId, recordId, maxResults, result)**

Register a `ConnectApi.TopicSuggestionPage` object to be returned when the matching `ConnectApi.getTopicSuggestions` method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**

29.0

**Signature**

```java
public static Void setTestGetTopicSuggestions(String communityId, String recordId,
Integer maxResults, ConnectApi.TopicSuggestionPage result)
```

**Parameters**

- `communityId`
  - Type: `String`
ID for an Experience Cloud site, internal, or null.

recordId
Type: String
ID for a record or feed item.

maxResults
Type: Integer
Maximum number of topic suggestions that get returned. The default is 5. Value must be greater than 0 and less than or equal to 25.

result
Type: ConnectApi.TopicSuggestionPage
Specify the test topic suggestions page.

Return Value
Type: Void

SEE ALSO:
- getTopicSuggestions(communityId, recordId, maxResults)

setTestGetTopicSuggestions(communityId, recordId, result)
Register a ConnectApi.TopicSuggestionPage object to be returned when the matching ConnectApi.getTopicSuggestions method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
29.0

Signature
public static Void setTestGetTopicSuggestions(String communityId, String recordId, ConnectApi.TopicSuggestionPage result)

Parameters
communityId
Type: String
ID for an Experience Cloud site, internal, or null.

recordId
Type: String
ID for a record or feed item.

result
Type: ConnectApi.TopicSuggestionPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
getTopicSuggestions(communityId, recordId)

setTestGetTopicSuggestionsForText(communityId, text, maxResults, result)
Register a ConnectApi.TopicSuggestionPage object to be returned when the matching ConnectApi.getTopicSuggestionsForText method is called in a test context. Use the method with the same parameters or you receive an exception.

API Version
29.0

Signature
public static Void setTestGetTopicSuggestionsForText(String communityId, String text, Integer maxResults, ConnectApi.TopicSuggestionPage result)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

text
Type: String
String of text.

maxResults
Type: Integer
Maximum number of topic suggestions that get returned. The default is 5. Value must be greater than 0 and less than or equal to 25.

result
Type: ConnectApi.TopicSuggestionPage
Object containing test data.
Return Value
Type: Void

SEE ALSO:
getTopicSuggestionsForText(communityId, text, maxResults)

**setTestGetTopicSuggestionsForText(communityId, text, result)**

Register a ConnectApi.TopicSuggestionPage object to be returned when the matching ConnectApi.getTopicSuggestionsForText method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**
29.0

**Signature**

```
public static Void setTestGetTopicSuggestionsForText(String communityId, String text, ConnectApi.TopicSuggestionPage result)
```

**Parameters**

- **communityId**
  Type: String
  ID for an Experience Cloud site, internal, or null.

- **text**
  Type: String
  String of text.

- **result**
  Type: ConnectApi.TopicSuggestionPage
  Object containing test data.

**Return Value**
Type: Void

SEE ALSO:
getTopicSuggestionsForText(communityId, text)

**setTestGetTrendingTopics(communityId, result)**

Register a ConnectApi.TopicPage object to be returned when the matching ConnectApi.getTrendingTopics method is called in a test context. Use the method with the same parameters or you receive an exception.
**setTestGetTrendingTopics(String communityId, ConnectApi.TopicPage result)**

**Parameters**

- **communityId**
  
  Type: String
  
  ID for an Experience Cloud site, internal, or null.

- **result**
  
  Type: ConnectApi.TopicPage
  
  Object containing test data.

**Return Value**

Type: Void

**SEE ALSO:**

- getTrendingTopics(communityId)

**setTestGetTrendingTopics(String communityId, maxResults, result)**

Register a ConnectApi.TopicPage object to be returned when the matching ConnectApi.getTrendingTopics method is called in a test context. Use the method with the same parameters or you receive an exception.

**API Version**

29.0

**Signature**

```java
public static Void setTestGetTrendingTopics(String communityId, Integer maxResults, ConnectApi.TopicPage result)
```

**Parameters**

- **communityId**
  
  Type: String
  
  ID for an Experience Cloud site, internal, or null.

- **maxResults**
  
  Type: Integer
Maximum number of topic suggestions that get returned. The default is 5. Value must be greater than 0 and less than or equal to 25.

result
Type: ConnectApi.TopicPage
Object containing test data.

Return Value
Type: Void

SEE ALSO:
getTrendingTopics(communityId, maxResults)

UserProfiles Class
Access user profile data. The user profile data populates the profile page (also called the Chatter profile page). This data includes user information (such as address, manager, and phone number), some user capabilities (permissions), and a set of subtab apps, which are custom tabs on the profile page.

Namespace
ConnectApi

UserProfiles Methods
The following are methods for UserProfiles. All methods are static.

IN THIS SECTION:

deleteBannerPhoto(communityId, userId)
Delete a user’s banner photo.

deletePhoto(communityId, userId)
Delete a user’s photo.

getBannerPhoto(communityId, userId)
Get a user’s banner photo.

getPhoto(communityId, userId)
Get a user’s photo.

getUserProfile(communityId, userId)
Get the user profile of the context user.

setBannerPhoto(communityId, userId, fileId, versionNumber)
Set an uploaded file as a user’s banner photo.

setBannerPhoto(communityId, userId, fileUpload)
Set a file that hasn’t been uploaded as a user’s banner photo.
**deleteBannerPhoto(communityId, userId)**

Delete a user's banner photo.

**API Version**

36.0

**Requires Chatter**

Yes

**Signature**

```java
public static Void deleteBannerPhoto(String communityId, String userId)
```

**Parameters**

- **communityId**
  - Type: **String**
  - ID for an Experience Cloud site, internal, or `null`.

- **userId**
  - Type: **String**
  - ID of the user.

**Return Value**

Type: **Void**

**deletePhoto(communityId, userId)**

Delete a user's photo.
API Version
35.0

Requires Chatter
Yes

Signature
public static Void deletePhoto(String communityId, String userId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

userId
Type: String
ID for the context user or the keyword me.

Return Value
Type: Void

getBannerPhoto(communityId, userId)
Get a user's banner photo.

API Version
36.0

Requires Chatter
Yes

Signature
public static ConnectApi.BannerPhoto getBannerPhoto(String communityId, String userId)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

userId
Type: String
ID of the user.
Return Value
Type: `ConnectApi.BannerPhoto`

`getPhoto(communityId, userId)`
Get a user's photo.

API Version
35.0

Available to Guest Users
35.0

Requires Chatter
Yes

Signature
public static ConnectApi.Photo getPhoto(String communityId, String userId)

Parameters
`communityId`
Type: `String`
ID for an Experience Cloud site, internal, or `null`.

`userId`
Type: `String`
ID for a user.

Return Value
Type: `ConnectApi.Photo`

`getUserProfile(communityId, userId)`
Get the user profile of the context user.

API Version
29.0

Requires Chatter
Yes
Signature

```java
public static ConnectApi.UserProfile getUserProfile(String communityId, String userId)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.
- `userId`
  - Type: `String`
  - ID for a user.

Return Value

Type: `ConnectApi.UserProfile`

`setBannerPhoto(communityId, userId, fileId, versionNumber)`

Set an uploaded file as a user's banner photo.

API Version

36.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.BannerPhoto setBannerPhoto(String communityId, String userId, String fileId, Integer versionNumber)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.
- `userId`
  - Type: `String`
  - ID of the user.
- `fileId`
  - Type: `String`
  - ID of the uploaded file to use as the user banner. The key prefix must be 069, and the image must be smaller than 8 MB.
- `versionNumber`
  - Type: `Integer`
Version number of the file. Specify an existing version number or, to get the latest version, specify null.

Return Value
Type: ConnectApi.BannerPhoto

Usage
Photos are processed asynchronously and might not be visible right away.

setBannerPhoto(communityId, userId, fileUpload)
Set a file that hasn’t been uploaded as a user’s banner photo.

API Version
36.0

Requires Chatter
Yes

Signature
public static ConnectApi.BannerPhoto setBannerPhoto(String communityId, String userId, ConnectApi.BinaryInput fileUpload)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

userId
Type: String
ID of the user.

fileUpload
Type: ConnectApi.BinaryInput
File to use as the photo. The content type must be usable as an image.

Return Value
Type: ConnectApi.BannerPhoto

Usage
Photos are processed asynchronously and might not be visible right away.
**setBannerPhotoWithAttributes(communityId, userId, bannerPhoto)**

Set and crop an uploaded file as a user’s banner photo.

**API Version**

36.0

**Requires Chatter**

Yes

**Signature**

```java
public static ConnectApi.BannerPhoto setBannerPhotoWithAttributes(String communityId,
String userId, ConnectApi.BannerPhotoInput bannerPhoto)
```

**Parameters**

- **communityId**
  - Type: String
  - ID for an Experience Cloud site, internal, or **null**.

- **userId**
  - Type: String
  - ID of the user.

- **bannerPhoto**
  - Type: ConnectApi.BannerPhotoInput
  - A `ConnectApi.BannerPhotoInput` object that specifies the ID and version of the file, and how to crop the file.

**Return Value**

Type: `ConnectApi.BannerPhoto`

**Usage**

Photos are processed asynchronously and might not be visible right away.

**setBannerPhotoWithAttributes(communityId, userId, bannerPhoto, fileUpload)**

Set and crop a file that hasn’t been uploaded as a user’s banner photo.

**API Version**

36.0

**Requires Chatter**

Yes
Signature

public static ConnectApi.BannerPhoto setBannerPhotoWithAttributes(String communityId, String userId, ConnectApi.BannerPhotoInput bannerPhoto, ConnectApi.BinaryInput fileUpload)

Parameters

communityId
  Type: String
  ID for an Experience Cloud site, internal, or null.

userId
  Type: String
  ID of the user.

bannerPhoto
  Type: ConnectApi.BannerPhotoInput
  A ConnectApi.BannerPhotoInput object specifying the cropping parameters.

fileUpload
  Type: ConnectApi.BinaryInput
  File to use as the photo. The content type must be usable as an image.

Return Value

Type: ConnectApi.BannerPhoto

Usage

Photos are processed asynchronously and might not be visible right away.

setPhoto(communityId, userId, fileId, versionNumber)

Set an uploaded file as a user’s photo.

API Version

35.0

Requires Chatter

Yes

Signature

public static ConnectApi.Photo setPhoto(String communityId, String userId, String fileId, Integer versionNumber)
Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

userId
Type: String
ID for the context user or the keyword me.

fileId
Type: String
ID of an uploaded file. The file must be an image, and be smaller than 2 GB.

versionNumber
Type: Integer
Version number of the existing file. Specify either an existing version number, or null to get the latest version.

Return Value
Type: ConnectApi.Photo

Usage
Photos are processed asynchronously and might not be visible right away.

setPhoto(communityId, userId, fileUpload)
Set a file that hasn’t been uploaded as a user’s photo.

API Version
35.0

Requires Chatter
Yes

Signature
public static ConnectApi.Photo setPhoto(String communityId, String userId, ConnectApi.BinaryInput fileUpload)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

userId
Type: String
ID for the context user or the keyword `me`.

`fileUpload`
Type: `ConnectApi.BinaryInput`
File to use as the photo. The content type must be usable as an image.

**Return Value**
Type: `ConnectApi.Photo`

**Usage**
Photos are processed asynchronously and might not be visible right away.

`setPhotoWithAttributes(communityId, userId, photo)`
Set and crop an uploaded file as a user’s photo.

**API Version**
35.0

**Requires Chatter**
Yes

**Signature**
```java
public static ConnectApi.Photo setPhotoWithAttributes(String communityId, String userId, ConnectApi.PhotoInput photo)
```

**Parameters**

- `communityId`
  Type: `String`
  ID for an Experience Cloud site, `internal`, or `null`.

- `userId`
  Type: `String`
  ID for the context user or the keyword `me`.

- `photo`
  Type: `ConnectApi.PhotoInput`
  A `ConnectApi.PhotoInput` object specifying the file ID, version number, and cropping parameters.

**Return Value**
Type: `ConnectApi.Photo`
Usage
Photos are processed asynchronously and might not be visible right away.

**setPhotoWithAttributes(communityId, userId, photo, fileUpload)**
Set and crop a file that hasn't been uploaded as a user's photo.

API Version
35.0

Requires Chatter
Yes

Signature
```
public static ConnectApi.Photo setPhotoWithAttributes(String communityId, String userId,
ConnectApi.PhotoInput photo, ConnectApi.BinaryInput fileUpload)
```

Parameters
- **communityId**
  - Type: *String*
  - ID for an Experience Cloud site, internal, or null.
- **userId**
  - Type: *String*
  - ID for the context user or the keyword me.
- **photo**
  - Type: *ConnectApi.PhotoInput*
  - A ConnectApi.PhotoInput object specifying the cropping parameters.
- **fileUpload**
  - Type: *ConnectApi.BinaryInput*
  - File to use as the photo. The content type must be usable as an image.

Return Value
Type: *ConnectApi.Photo*

Usage
Photos are processed asynchronously and might not be visible right away.

Zones Class
Access information about Chatter Answers zones in your organization. Zones organize questions into logical groups, with each zone having its own focus and unique questions.
Note: With the Spring ’18 release, Salesforce no longer supports Chatter Answers. Users of Chatter Answers can post, answer, comment, or view existing Chatter Answers data, but support and updates are scheduled to end. We recommend transitioning to Chatter Questions. For more information, see End of Support for Chatter Answers in Spring ’18.

Namespace

ConnectApi

Zones Methods

The following are methods for Zones. All methods are static.

IN THIS SECTION:

getZone(communityId, zoneId)
Get a zone.

getZones(communityId)
Get a list of zones.

getZones(communityId; pageParam, pageSize)
Get a page of zones.

searchInZone(communityId, zoneId, q, filter)
Search articles or questions in a zone.

searchInZone(communityId, zoneId, q, filter, pageParam, pageSize)
Search a page of articles or questions in a zone.

searchInZone(communityId, zoneId, q, filter, language)
Search articles or questions in a zone, and specify the language of the results.

getZone(communityId, zoneId)
Get a zone.

Note: With the Spring ’18 release, Salesforce no longer supports Chatter Answers. Users of Chatter Answers can post, answer, comment, or view existing Chatter Answers data, but support and updates are scheduled to end. We recommend transitioning to Chatter Questions. For more information, see End of Support for Chatter Answers in Spring ’18.

API Version

29.0

Requires Chatter

Yes

Signature

public static ConnectApi.Zone getZone(String communityId, String zoneId)
Parameters

`communityId`
Type: String
ID for an Experience Cloud site, internal, or null.

`zoneId`
Type: String
The ID of a zone.

Return Value

Type: `ConnectApi.Zone`

**getZones (communityId)**

Get a list of zones.

⚠️ Note: With the Spring '18 release, Salesforce no longer supports Chatter Answers. Users of Chatter Answers can post, answer, comment, or view existing Chatter Answers data, but support and updates are scheduled to end. We recommend transitioning to Chatter Questions. For more information, see End of Support for Chatter Answers in Spring '18.

API Version

29.0

Requires Chatter

Yes

Signature

```java
public static ConnectApi.ZonePage getZones(String communityId)
```

Parameters

`communityId`
Type: String
ID for an Experience Cloud site, internal, or null.

Return Value

Type: `ConnectApi.ZonePage`

**getZones (communityId, pageParam, pageSize)**

Get a page of zones.

⚠️ Note: With the Spring '18 release, Salesforce no longer supports Chatter Answers. Users of Chatter Answers can post, answer, comment, or view existing Chatter Answers data, but support and updates are scheduled to end. We recommend transitioning to Chatter Questions. For more information, see End of Support for Chatter Answers in Spring '18.
API Version
29.0

Requires Chatter
Yes

Signature
```
public static ConnectApi.Zone getZones(String communityId, Integer pageParam, Integer pageSize)
```

Parameters
- `communityId`  
  Type: String  
  ID for an Experience Cloud site, internal, or null.
- `pageParam`  
  Type: Integer  
  Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.
- `pageSize`  
  Type: Integer  
  Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

Return Value
Type: `ConnectApi.ZonePage`

`searchInZone(communityId, zoneId, q, filter)`
Search articles or questions in a zone.

Note: With the Spring ’18 release, Salesforce no longer supports Chatter Answers. Users of Chatter Answers can post, answer, comment, or view existing Chatter Answers data, but support and updates are scheduled to end. We recommend transitioning to Chatter Questions. For more information, see End of Support for Chatter Answers in Spring ’18.
Signature

```java
public static ConnectApi.ZoneSearchPage searchInZone(String communityId, String zoneId, String q, ConnectApi.ZoneSearchResultType filter)
```

Parameters

- `communityId`
  - Type: `String`
  - ID for an Experience Cloud site, internal, or `null`.

- `zoneId`
  - Type: `String`
  - ID of a zone.

- `q`
  - Type: `String`
  - Specifies the string to search. The search string must contain at least two characters, not including wildcards. See [Wildcards](#).

- `filter`
  - Type: `ConnectApi.ZoneSearchResultType`
  - A `ZoneSearchResultType` enum value. One of the following:
    - Article—Search results contain only articles.
    - Question—Search results contain only questions.

Return Value

- Type: `ConnectApi.ZoneSearchPage`

Usage

To test code that uses this method, use the matching set test method (prefix the method name with `setTest`). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:

- `setTestSearchInZone(communityId, zoneId, q, filter, result)`
- *Apex Developer Guide: Testing ConnectApi Code*

**searchInZone(communityId, zoneId, q, filter, pageParam, pageSize)**

Search a page of articles or questions in a zone.

**Note:** With the Spring ’18 release, Salesforce no longer supports Chatter Answers. Users of Chatter Answers can post, answer, comment, or view existing Chatter Answers data, but support and updates are scheduled to end. We recommend transitioning to Chatter Questions. For more information, see [End of Support for Chatter Answers in Spring ’18](#).

API Version

- 29.0
Available to Guest Users
37.0

Requires Chatter
Yes

Signature
public static ConnectApi.ZoneSearchPage searchInZone(String communityId, String zoneId, String q, ConnectApi.ZoneSearchResultType filter, String pageParam, Integer pageSize)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

zoneId
Type: String
ID of a zone.

q
Type: String
Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

filter
Type: ConnectApi.ZoneSearchResultType
A ZoneSearchResultType enum value. One of the following:
• Article—Search results contain only articles.
• Question—Search results contain only questions.

pageParam
Type: String
Specifies the page token to use to view a page of information. Page tokens are returned as part of the response class, such as currentPageToken or nextPageToken. If you pass in null, the first page is returned.

pageSize
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in null, the default size is 25.

Return Value
Type: ConnectApi.ZoneSearchPage
Usage
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
  setTestSearchInZone(communityId, zoneId, q, filter, pageParam, pageSize, result)

searchInZone(communityId, zoneId, q, filter, language)
Search articles or questions in a zone, and specify the language of the results.

Note: With the Spring '18 release, Salesforce no longer supports Chatter Answers. Users of Chatter Answers can post, answer, comment, or view existing Chatter Answers data, but support and updates are scheduled to end. We recommend transitioning to Chatter Questions. For more information, see End of Support for Chatter Answers in Spring '18.

API Version
36.0

Available to Guest Users
37.0

Requires Chatter
Yes

Signature
public static ConnectApi.ZoneSearchPage searchInZone(String communityId, String zoneId, String q, ConnectApi.ZoneSearchResultType filter, String language)

Parameters

communityId
Type: String
ID for an Experience Cloud site, internal, or null.

zoneId
Type: String
ID of a zone.

q
Type: String
Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

filter
Type: ConnectApi.ZoneSearchResultType
- Article—Search results contain only articles.
• Question—Search results contain only questions.

**language**
Type: String

The language of the articles or questions. The value must be a Salesforce supported locale code.

**Return Value**
Type: ConnectApi.ZoneSearchPage

**Usage**
To test code that uses this method, use the matching set test method (prefix the method name with setTest). Use the set test method with the same parameters or the code throws an exception.

SEE ALSO:
setTestSearchInZone(communityId, zoneId, q, filter, language, result)

**Zones Test Methods**
The following are the test methods for Zones. All methods are static.
For information about using these methods to test your ConnectApi code, see Testing ConnectApi Code.

**setTestSearchInZone(communityId, zoneId, q, filter, result)**
Register a ConnectApi.ZoneSearchPage object to be returned when searchInZone(communityId, zoneId, q, filter) is called in a test context. Use the method with the same parameters or you receive an exception.

⚠️ Note: With the Spring '18 release, Salesforce no longer supports Chatter Answers. Users of Chatter Answers can post, answer, comment, or view existing Chatter Answers data, but support and updates are scheduled to end. We recommend transitioning to Chatter Questions. For more information, see End of Support for Chatter Answers in Spring '18.

**API Version**
29.0

**Signature**
public static Void setTestSearchInZone(String communityId, String zoneId, String q, ConnectApi.ZoneSearchResultType filter, ConnectApi.ZoneSearchPage result)

**Parameters**

- **communityId**
  Type: String
  Use either the ID for an Experience Cloud site, internal, or null.

- **zoneId**
  Type: String
  The ID of a zone.
q
  Type: String
  Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

filter
  Type: ConnectApi.ZoneSearchResultType
  A ZoneSearchResultType enum value. One of the following:
  • Article—Search results contain only articles.
  • Question—Search results contain only questions.

result
  Type: ConnectApi.ZoneSearchPage
  The object containing test data.

Return Value
Type: Void

SEE ALSO:
  searchInZone(communityId, zoneId, q, filter)

setTestSearchInZone(communityId, zoneId, q, filter, pageParam, pageSize, result)
Register a ConnectApi.ZoneSearchPage object to be returned when searchInZone(communityId, zoneId, q, filter, pageParam, pageSize) is called in a test context. Use the method with the same parameters or you receive an exception.

Note: With the Spring '18 release, Salesforce no longer supports Chatter Answers. Users of Chatter Answers can post, answer, comment, or view existing Chatter Answers data, but support and updates are scheduled to end. We recommend transitioning to Chatter Questions. For more information, see End of Support for Chatter Answers in Spring '18.

API Version
29.0

Signature
public static Void setTestSearchInZone(String communityId, String zoneId, String q, ConnectApi.ZoneSearchResultType filter, String pageParam, Integer pageSize, ConnectApi.ZoneSearchPage result)

Parameters
communityId
  Type: String
  Use either the ID for an Experience Cloud site, internal, or null.
**zoneId**
Type: String
The ID of a zone.

**q**
Type: String
Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

**filter**
Type: `ConnectApi.ZoneSearchResultType`
A `ZoneSearchResultType` enum value. One of the following:
- Article—Search results contain only articles.
- Question—Search results contain only questions.

**pageParam**
Type: String
Specifies the page token to be used to view a page of information. Page tokens are returned as part of the response class, such as `currentPageToken` or `nextPageToken`. If you pass in `null`, the first page is returned.

**pageSize**
Type: Integer
Specifies the number of items per page. Valid values are from 1 through 100. If you pass in `null`, the default size is 25.

**result**
Type: `ConnectApi.ZoneSearchPage`
The object containing test data.

**Return Value**
Type: Void

**SEE ALSO:**
- `searchInZone(communityId, zoneId, q, filter, pageParam, pageSize)`

**setTestSearchInZone(communityId, zoneId, q, filter, language, result)**
Register a `ConnectApi.ZoneSearchPage` object to be returned when `searchInZone(communityId, zoneId, q, filter, language)` is called in a test context. Use the method with the same parameters or you receive an exception.

**Note:** With the Spring ’18 release, Salesforce no longer supports Chatter Answers. Users of Chatter Answers can post, answer, comment, or view existing Chatter Answers data, but support and updates are scheduled to end. We recommend transitioning to Chatter Questions. For more information, see End of Support for Chatter Answers in Spring ’18.

**API Version**
36.0
public static Void setTestSearchInZone(String communityId, String zoneId, String q, ConnectApi.ZoneSearchResultType filter, String language, ConnectApi.ZoneSearchPage result)

Parameters

communityId
Type: String
Use either the ID for an Experience Cloud site, internal, or null.

zoneId
Type: String
The ID of a zone.

q
Type: String
Specifies the string to search. The search string must contain at least two characters, not including wildcards. See Wildcards.

filter
Type: ConnectApi.ZoneSearchResultType
- Article—Search results contain only articles.
- Question—Search results contain only questions.

language
Type: String
The language of the articles or questions. The value must be a Salesforce supported locale code. In an <apex:page>, the default value is the language of the page. Otherwise, the default value is the user’s locale.

result
Type: ConnectApi.ZoneSearchPage
The object containing test data.

Return Value
Type: Void

SEE ALSO:
- searchInZone(communityId, zoneId, q, filter, language)

ConnectApi Input Classes

Some ConnectApi methods take arguments that are instances of ConnectApi input classes. Input classes are concrete unless marked abstract in this documentation. Concrete input classes have public constructors that have no parameters. Some methods have parameters that are typed with an abstract class. You must pass in an instance of a concrete child class for these parameters.
Most input class properties can be set. Read-only properties are noted in this documentation.

**ConnectApi.ActionLinkDefinitionInput**

The definition of an action link. An action link is a button on a feed element. Clicking an action link can take a user to a Web page, initiate a file download, or invoke an API call to Salesforce or to an external server. An action link includes a URL and an HTTP method, and can include a request body and header information, such as an OAuth token for authentication. Use action links to integrate Salesforce and third-party services into the feed so that users can drive productivity and accelerate innovation.

**Usage**

You can use context variables in the `actionUrl`, `headers`, and `requestBody` properties. Use context variables to pass information about the user who executed the action link to your server-side code. Salesforce substitutes the value when the action link is executed.

The available context variables are:

<table>
<thead>
<tr>
<th>Context Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{!actionLinkId}</td>
<td>The ID of the action link the user executed.</td>
</tr>
<tr>
<td>{!actionLinkGroupId}</td>
<td>The ID of the action link group containing the action link the user executed.</td>
</tr>
<tr>
<td>{!communityId}</td>
<td>The ID of the site in which the user executed the action link. The value for your internal org is the empty key &quot;000000000000000000&quot;.</td>
</tr>
<tr>
<td>{!communityUrl}</td>
<td>The URL of the site in which the user executed the action link. The value for your internal org is empty string &quot;&quot;.</td>
</tr>
<tr>
<td>{!orgId}</td>
<td>The ID of the org in which the user executed the action link.</td>
</tr>
<tr>
<td>{!userId}</td>
<td>The ID of the user that executed the action link.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>actionType</td>
<td><code>ConnectApi.ActionLinkType</code></td>
<td>Defines the type of action link. Values are:</td>
<td>Required</td>
<td>33.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- <strong>Api</strong>—The action link calls a synchronous API at the action URL. Salesforce sets the status to <code>SuccessfulStatus</code> or <code>FailedStatus</code> based on the HTTP status code returned by your server.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
|                 |                             | - **ApiAsync**—The action link calls an asynchronous API at the action URL. The action remains in a `PendingStatus` state until a third party makes a request to `/connect/action-links/{actionLinkId}` to set the status to
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>actionUrl</td>
<td>String</td>
<td>The action link URL. For example, a Ui action link URL is a Web page. A Download action link URL is a link to the file to download. Ui and Download action link URLs are provided to clients. An Api or ApiAsync action link URL is a REST resource. Api and ApiAsync action link URLs aren’t provided to clients. Links to Salesforce can be relative. All other links must be absolute and start with https://.</td>
<td>Required</td>
<td>33.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Tip: To avoid issues due to upgrades or changing functionality in your API, we recommend using a versioned API for actionUrl, for example, <a href="https://www.example.com/api/v1/exampleResource">https://www.example.com/api/v1/exampleResource</a>. If your API isn’t versioned, you can use the expirationDate property of the ConnectApi.ActionLinkGroup DefinitionInput class to</td>
<td>Can be defined in an action link template.</td>
<td></td>
</tr>
<tr>
<td>Property</td>
<td>Type</td>
<td>Description</td>
<td>Required or Optional</td>
<td>Available Version</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>excludedUserId</td>
<td>String</td>
<td>ID of a single user to exclude from performing the action. If you specify an excludedUserId, you can’t specify a userId.</td>
<td>Optional</td>
<td>33.0</td>
</tr>
<tr>
<td>groupDefault</td>
<td>Boolean</td>
<td>true if this action is the default action link in the action link group; false otherwise. There can be only one default action link per action link group. The default action link gets distinct styling in the Salesforce UI.</td>
<td>Optional</td>
<td>33.0</td>
</tr>
<tr>
<td>headers</td>
<td>List&lt;ConnectApi.RequestHeaderInput&gt;</td>
<td>The request headers for the Api and ApiAsync action link types. See Action Links Overview, Authentication, and Security.</td>
<td>Optional</td>
<td>33.0</td>
</tr>
<tr>
<td>labelKey</td>
<td>String</td>
<td>Key for the set of labels to show in the user interface. A set includes labels for these states: NewStatus, PendingStatus, SuccessStatus, FailedStatus. For example, if you use the Approve key, you get these labels: Approve, Pending, Approved, Failed. For a complete list of keys and labels, see Action Links Labels. If none of the predefined labels work for your action link, use a custom label. To use a custom label, create an action link template. See Create Action Link Templates.</td>
<td>Required</td>
<td>33.0</td>
</tr>
</tbody>
</table>
| method             | ConnectApi.HttpRequestMethod | One of these HTTP methods:  
- `HttpDelete`—Returns HTTP 204 on success. Response body or output class is empty.  
- `HttpGet`—Returns HTTP 200 on success.  
- `HttpHead`—Returns HTTP 200 on success. Response body or output class is empty. | Required             | 33.0              |
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>requestBody</td>
<td>String</td>
<td>The request body for API action links.</td>
<td>Optional</td>
<td>33.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Note: Escape quotation mark characters in the <code>requestBody</code> value.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>requires Confirmation</td>
<td>Boolean</td>
<td>true to require the user to confirm the action; false otherwise.</td>
<td>Required</td>
<td>33.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Can be defined in an action link template.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>userId</td>
<td>String</td>
<td>The ID of the user who can execute the action. If not specified or null, any user can execute the action. If you specify a <code>userId</code>, you can't specify an <code>excludedUserId</code>.</td>
<td>Optional</td>
<td>33.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Can be defined in an action link template using the User Visibility and Custom User Alias fields.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

See ALSO:

`ConnectApi.ActionLinkGroupDefinitionInput`

**ConnectApi.ActionLinkGroupDefinitionInput**

The definition of an action link group. All action links must belong to a group. Action links in a group are mutually exclusive and share some properties. Define standalone actions in their own action group.

Action link definition can be sensitive to a third party (for example, OAuth bearer token headers). For this reason, only calls made from the Apex namespace that created the action link definition can read, modify, or delete the definition. In addition, the user making the call must have created the definition or have View All Data permission.
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>actionLinks</td>
<td><code>List&lt;ConnectApi.ActionLink DefinitionInput&gt;</code></td>
<td>The action links that make up this group. Within an action link group, action links are displayed in the order listed in the <code>actionLinks</code> property of the <code>ConnectApi.ActionLinkGroup DefinitionInput</code> class. Within a feed item, action link groups are displayed in the order specified in the <code>actionLinkGroupIds</code> property of the <code>ConnectApi.AssociatedActions CapabilityInput</code> class. You can create up to three action links in a Primary group and up to four in an Overflow group.</td>
<td>Required to instantiate this action link group without a template. To instantiate from a template, don’t specify a value.</td>
<td>33.0</td>
</tr>
<tr>
<td>category</td>
<td><code>ConnectApi.PlatformAction GroupCategory</code></td>
<td>Indicates the priority and relative locations of action links in an associated feed item. Values are:</td>
<td>Required to instantiate this action link group without a template. To instantiate from a template, don’t specify a value.</td>
<td>33.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Primary—The action link group is displayed in the body of the feed element.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Overflow—The action link group is displayed in the overflow menu of the feed element.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>executions Allowed</td>
<td><code>ConnectApi.ActionLink ExecutionsAllowed</code></td>
<td>Defines the number of times an action link can be executed. Values are:</td>
<td>Required to instantiate this action link group without a template. To instantiate from a template, don’t specify a value.</td>
<td>33.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Once—An action link can be executed only one time across all users.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- OncePerUser—An action link can be executed only one time for each user.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Unlimited—An action link can be executed an unlimited number of times by each user. If the action link’s <code>actionType</code> is <code>Api</code> or <code>ApiAsync</code>, you can’t use this value.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>expirationDate</td>
<td><code>Datetime</code></td>
<td>ISO 8601 date string, for example, 2011-02-25T18:24:31.000Z, that represents</td>
<td>Required to instantiate this action link group without a template.</td>
<td>33.0</td>
</tr>
</tbody>
</table>
the date and time this action link group is removed from associated feed items and can no longer be executed. The expirationDate must be within one year of the creation date.

If the action link group definition includes an OAuth token, it is a good idea to set the expiration date of the action link group to the same value as the expiration date of the OAuth token so that users can’t execute the action link and get an OAuth error.

To set a date when instantiating from a template, see Set the Action Link Group Expiration Time in Design Action Link Templates.

To instantiate without a template, don’t specify a value.

Required to instantiate this action link group from a template.

A collection of key-value pairs to fill in binding variable values or a custom user alias from an action link template. To instantiate this action link group from an action link template that uses binding variables, you must provide values for all the variables. See Define Binding Variables in Design Action Link Templates.

A key-value pair to fill in a binding variable value from an action link template.

SEE ALSO:

Define an Action Link and Post with a Feed Element
Define an Action Link in a Template and Post with a Feed Element
createActionLinkGroupDefinition(communityId, actionLinkGroup)
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>key</td>
<td>String</td>
<td>The name of the binding variable key specified in the action link template in Setup. For example, if the binding variable in the template is <code>{!Binding.firstName}</code>, the key is firstName.</td>
<td>Required</td>
<td>33.0</td>
</tr>
<tr>
<td>value</td>
<td>String</td>
<td>The value of the binding variable key. For example, if the key is firstName, this value could be Joan.</td>
<td>Required</td>
<td>33.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.ActionLinkGroupDefinitionInput

**ConnectApi.ActivitySharingInput**

Defines who a captured email or event is shared with.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>groupsToShareWith</td>
<td>List&lt;String&gt;</td>
<td>List of IDs for the groups that you share the activity with. Valid only if sharingType is MyGroups.</td>
<td>Optional</td>
<td>39.0</td>
</tr>
</tbody>
</table>
| sharingType      | ConnectApi.ActivitySharingType | Type of sharing operation. Values are:  
  - Everyone—The activity is shared with everyone.  
  - MyGroups—The activity is shared only with a selection of the context user’s groups.  
  - OnlyMe—The activity is private. | Required             | 39.0              |

**ConnectApi.AddressRequest**

Address input representation for a payment method or card payment method.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>city</td>
<td>String</td>
<td>Payment method city.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
<tr>
<td>companyName</td>
<td>String</td>
<td>Payment method company name.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
<tr>
<td>country</td>
<td>String</td>
<td>Payment method country.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
<td>Required or Optional</td>
<td>Available Version</td>
</tr>
<tr>
<td>-------------------</td>
<td>--------</td>
<td>---------------------------</td>
<td>----------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>postalCode</td>
<td>String</td>
<td>Payment method postal code.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
<tr>
<td>state</td>
<td>String</td>
<td>Payment method state.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
<tr>
<td>street</td>
<td>String</td>
<td>Payment method street.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
</tbody>
</table>

**ConnectApi.AdjustItemInputRepresentation**

A price adjustment to an OrderItemSummary. It only supports discounts, not increases.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>adjustmentType</td>
<td>String</td>
<td>Describes how the amount is calculated. It can have one of these values:</td>
<td>Required</td>
<td>49.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AmountWithTax—Value of amount is the adjustment, including tax.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AmountWithoutTax—Value of amount is the adjustment, not including tax.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Tax is calculated on the value and added.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Percentage—Value of amount is a percentage discount. It is divided by</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>100, and then multiplied by the TotalPrice and TotalTaxAmount of the</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>OrderItemSummary to determine the adjustment amount.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>amount</td>
<td>Double</td>
<td>Value used to calculate the adjustment amount, as described by the</td>
<td>Required</td>
<td>49.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>adjustmentType. It must be a negative value.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the adjustment.</td>
<td>Optional</td>
<td>49.0</td>
</tr>
<tr>
<td>orderItemSummaryId</td>
<td>String</td>
<td>ID of the OrderItemSummary.</td>
<td>Required</td>
<td>49.0</td>
</tr>
<tr>
<td>reason</td>
<td>String</td>
<td>Reason for the adjustment. The value must match one of the picklist values</td>
<td>Required</td>
<td>49.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>on the Reason field of the OrderItemSummaryChange object.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**ConnectApi.AdjustOrderItemSummaryInputRepresentation**

Price adjustments to OrderItemSummaries that together make up a price adjustment to an order, with options for adjusting items in the process of being fulfilled.
### Available Version

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>adjustItems</td>
<td>List of price adjustments to OrderItemSummaries.</td>
<td>Required</td>
<td>49.0</td>
</tr>
<tr>
<td>allocatedItems</td>
<td>Process to use for OrderItemSummary quantities that are currently being fulfilled, defined as ( \text{QuantityAllocated} - \text{QuantityFulfilled} ). Values are:</td>
<td>Optional</td>
<td>55.0</td>
</tr>
</tbody>
</table>
| ChangeOrderType | • Disallowed—When distributing the adjustment, ignore any quantities being fulfilled. If an OrderItemSummary’s entire quantity is being fulfilled, return an error. This is the default value.  
• InFulfillment—When distributing the adjustment, include quantities being fulfilled. Create a separate change order for the adjustments made to those quantities.  
• PreFulfillment—When distributing the adjustment, include quantities being fulfilled. Include the adjustments made to those quantities in the change order for pre-fulfillment quantity adjustments. | | |

### ConnectApi.AlternativeInput

Alternative representation for an extension on a feed element.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>text</td>
<td>Text representation of the extension.</td>
<td>Required</td>
<td>40.0</td>
</tr>
<tr>
<td>thumbnailUrl</td>
<td>Thumbnail URL to the extension.</td>
<td>Optional</td>
<td>40.0</td>
</tr>
<tr>
<td>title</td>
<td>Title of the extension.</td>
<td>Optional</td>
<td>40.0</td>
</tr>
</tbody>
</table>

### ConnectApi.AnnouncementInput

An announcement.
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available</th>
</tr>
</thead>
<tbody>
<tr>
<td>body</td>
<td>ConnectApi.MessageBodyInput</td>
<td>Text of the announcement.</td>
<td>Required</td>
<td>31.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Optional</td>
<td></td>
</tr>
<tr>
<td>expirationDate</td>
<td>Datetime</td>
<td>The Salesforce UI displays an announcement until 11:59 p.m. on this date unless another announcement is posted first. The Salesforce UI ignores the time value in the expirationDate. However, you can use the time value to create your own display logic in your own UI.</td>
<td>Required for creating an announcement</td>
<td>31.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Optional for updating an announcement</td>
<td></td>
</tr>
<tr>
<td>feedItemId</td>
<td>String</td>
<td>ID of an AdvancedTextPost feed item that is the body of the announcement.</td>
<td>Required</td>
<td>36.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Optional</td>
<td></td>
</tr>
<tr>
<td>isArchived</td>
<td>Boolean</td>
<td>Specifies whether the announcement is archived.</td>
<td>Optional</td>
<td>36.0</td>
</tr>
<tr>
<td>parentId</td>
<td>String</td>
<td>ID of the parent entity for the announcement, that is, a group ID when the announcement appears in a group.</td>
<td>Required</td>
<td>36.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Optional</td>
<td></td>
</tr>
</tbody>
</table>
**sendEmails**  
*Boolean*  
Specifies whether the announcement is sent as an email to all group members regardless of their email setting for the group. If Chatter emails aren’t enabled for the organization, announcement emails aren’t sent. Default value is *false*.

Optional for creating an announcement
Don’t specify for updating an announcement

### ConnectApi.ArticleTopicAssignmentJobInput

An article and topic assignment job.

**operation**  
*ConnectApi.ArticleTopicJobType*  
Type of operation to perform on articles and topics. Values are:
- AssignTopicsToArticle—Assign topics to articles in a data category.
- UnassignTopicsFromArticle—Unassign topics from articles in a data category.

**topicNames**  
*ConnectApi.TopicNamesInput*  
List of topic names to assign to or unassign from articles.

### ConnectApi.AssociatedActionsCapabilityInput

A list of action link groups to associate with a feed element. To associate an action link group with a feed element, the call must be made from the Apex namespace that created the action link definition. In addition, the user making the call must have created the definition or have View All Data permission.

An action link is a button on a feed element. Clicking an action link can take a user to a Web page, initiate a file download, or invoke an API call to Salesforce or to an external server. An action link includes a URL and an HTTP method, and can include a request body and header information, such as an OAuth token for authentication. Use action links to integrate Salesforce and third-party services into the feed so that users can drive productivity and accelerate innovation.

**SEE ALSO:**  
`postAnnouncement(communityId, groupId, announcement)`
### ConnectApi.AuditCommitInput

Custom audit commit criteria type.

This class is abstract and has no public constructor. You can make an instance only of a subclass.

**Superclass for:**
- ConnectApi.AuditCustomEmailAddressInput
- ConnectApi.AuditCustomEmailAddressInput
- ConnectApi.AuditCustomIdentityInput
- ConnectApi.AuditCustomPhoneInput

### ConnectApi.AuditEmailInput

Custom audit email criteria type.

This class is abstract and has no public constructor. You can make an instance only of a subclass.

**Superclass for:**
- ConnectApi.AuditCustomEmailAddressInput
- ConnectApi.AuditCustomEmailAddressInput
- ConnectApi.AuditCustomIdentityInput
- ConnectApi.AuditCustomPhoneInput

### ConnectApi.AuditIdentityInput

Custom audit identity criteria type.

This class is abstract and has no public constructor. You can make an instance only of a subclass.

**Superclass for:**
- ConnectApi.AuditCustomEmailAddressInput
- ConnectApi.AuditCustomEmailAddressInput
- ConnectApi.AuditCustomIdentityInput
- ConnectApi.AuditCustomPhoneInput

### ConnectApi.AuditPhoneInput

Custom audit phone criteria type.

This class is abstract and has no public constructor. You can make an instance only of a subclass.

**Superclass for:**
- ConnectApi.AuditCustomEmailAddressInput
- ConnectApi.AuditCustomEmailAddressInput
- ConnectApi.AuditCustomIdentityInput
- ConnectApi.AuditCustomPhoneInput

### ConnectApi.AuditPhoneInput

Custom audit phone criteria type.

This class is abstract and has no public constructor. You can make an instance only of a subclass.

**Superclass for:**
- ConnectApi.AuditCustomEmailAddressInput
- ConnectApi.AuditCustomEmailAddressInput
- ConnectApi.AuditCustomIdentityInput
- ConnectApi.AuditCustomPhoneInput

---

### ConnectApi.Input Classes

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>actionLinkGroupIds</td>
<td>List&lt;String&gt;</td>
<td>The action link group IDs to associate with the feed element. Associate one Primary and up to 10 total action link groups to a feed item. Action link groups are returned in the order specified in this property. An action link group ID is returned from a call to <code>ConnectApi.ActionLinks.createActionLinkGroupDefinition(communityId, actionLinkGroup)</code>.</td>
<td>Required</td>
<td>33.0</td>
</tr>
<tr>
<td>associateRecordIds</td>
<td>List&lt;String&gt;</td>
<td>ID of the associated records.</td>
<td>Required</td>
<td>50.0</td>
</tr>
<tr>
<td>recipientId</td>
<td>String</td>
<td>Participant ID with whose invitation the record should be associated.</td>
<td>Required</td>
<td>50.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

- ConnectApi.FeedElementCapabilitiesInput
- ConnectApi.AssociateRecordsWithRecipientInput
- ConnectApi.SurveyInvitationEmailInput
- ConnectApi.AudienceCriteriaInput

---

**ConnectApi.AssociateRecordsWithRecipientInput**

Records associated with the survey invitation.

**ConnectApi.AudienceCriteriaInput**

Custom recommendation audience criteria type.

This class is abstract and has no public constructor. You can make an instance only of a subclass.

**Superclass for:**
- ConnectApi.CustomListAudienceCriteriaInput
- ConnectApi.NewUserAudienceCriteriaInput
### Property | Type | Description | Required or Optional | Available Version
--- | --- | --- | --- | ---
**type** | `ConnectApi.RecommendationAudienceCriteriaType` | Specifies the custom recommendation audience criteria type. One of these values:
- CustomList—A custom list of users makes up the audience.
- MaxDaysInCommunity—New members make up the audience. | Optional | 36.0

**SEE ALSO:**
- `ConnectApi.RecommendationAudienceInput`

### `ConnectApi.AudienceCriterionInput`
Personalization audience criterion.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>criterion</strong></td>
<td><code>List&lt;ConnectApi.AudienceCriterionValueInput&gt;</code></td>
<td>List of mappings of audience criteria fields and values.</td>
<td>Required when creating an audience, Optional when updating an audience</td>
<td>48.0</td>
</tr>
<tr>
<td><strong>criterionNumber</strong></td>
<td><code>Integer</code></td>
<td>Number associated with the audience criterion in a formula. For example, (1 AND 2) OR 3. If unspecified, criteria are assigned numbers in the order that they're added.</td>
<td>Optional</td>
<td>48.0</td>
</tr>
</tbody>
</table>
| **criterionOperator** | `ConnectApi.AudienceCriteriaOperator` | Operator used in the personalization audience criterion. Values are:
- Contains
- Equal
- GreaterThan
- GreaterThanOrEqual
- Included
- LessThan
- LessThanOrEqual
- NotEqual
- NotIncludes
- StartsWith | Required when creating an audience, Optional when updating an audience | 48.0 |
### ConnectApi.AudienceCriterionValueInput

Audience criterion value.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>audienceId</td>
<td>String</td>
<td>ID of an audience.</td>
<td>Required if creating or updating an audience with the Audience criterion type.</td>
<td>53.0</td>
</tr>
<tr>
<td>city</td>
<td>String</td>
<td>City of a user.</td>
<td>Optional if creating or updating an audience with the GeoLocation criterion type</td>
<td>48.0</td>
</tr>
<tr>
<td>country</td>
<td>String</td>
<td>Country of a user.</td>
<td>Required if creating or updating an audience with the GeoLocation criterion type</td>
<td>48.0</td>
</tr>
<tr>
<td>domainId</td>
<td>String</td>
<td>Domain ID of a user.</td>
<td>Required if creating or updating an audience with the</td>
<td>48.0</td>
</tr>
<tr>
<td>Property</td>
<td>Type</td>
<td>Description</td>
<td>Required or Optional</td>
<td>Available Version</td>
</tr>
<tr>
<td>-------------------</td>
<td>----------</td>
<td>--------------------------------------------------</td>
<td>----------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>entityField</td>
<td>String</td>
<td>Field of an object.</td>
<td>Required if creating or updating an audience with the FieldBased criterion type</td>
<td>48.0</td>
</tr>
<tr>
<td>entityType</td>
<td>String</td>
<td>Type of object.</td>
<td>Required if creating or updating an audience with the FieldBased criterion type</td>
<td>48.0</td>
</tr>
<tr>
<td>fieldValue</td>
<td>String</td>
<td>Value of a field.</td>
<td>Required if creating or updating an audience with the FieldBased criterion type</td>
<td>48.0</td>
</tr>
<tr>
<td>isEnabled</td>
<td>Boolean</td>
<td>Specifies whether the permission is enabled (true) or not (false) for a user.</td>
<td>Required if creating or updating an audience with the Permission criterion type</td>
<td>48.0</td>
</tr>
<tr>
<td>permission</td>
<td>String</td>
<td>Valid API name of a standard user or custom permission.</td>
<td>Required if creating or updating an audience with the Permission criterion type</td>
<td>48.0</td>
</tr>
<tr>
<td>profileId</td>
<td>String</td>
<td>Profile ID of a user.</td>
<td>Required if creating or updating an audience with the Profile criterion type</td>
<td>48.0</td>
</tr>
</tbody>
</table>
### Available Version

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>subdivision</td>
<td>String</td>
<td>Subdivision of a user.</td>
<td>Required if creating or updating an audience with the GeoLocation criterion type and using the city property.</td>
<td>48.0</td>
</tr>
</tbody>
</table>

SEE ALSO:  
ConnectApi.AudienceCriterionInput

### ConnectApi.AudienceInput

A personalization audience.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>criteria</td>
<td>List&lt;ConnectApi.AudienceCriterionInput&gt;</td>
<td>List of audience criteria to update or add. An audience can have up to 100 criteria.</td>
<td>Required when creating an audience. Optional when updating an audience.</td>
<td>48.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>customFormula</td>
<td>String</td>
<td>Custom formula for the audience criteria. For example, (1 AND 2) OR 3.</td>
<td>Required when creating an audience with the formulaFilterType set to CustomLogicMatches. Optional, otherwise.</td>
<td>48.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
</table>
| formulaFilterType | ConnectApi.FormulaFilterType | Formula filter type for the personalization audience. Values are:  
• AllCriteriaMatch—All audience criteria are true (AND operation).  
• AnyCriterionMatches—Any audience criterion is true (OR operation).  
• CustomLogicMatches—Audience criteria match the custom formula (for example, (1 AND 2) OR 3). | Required when creating an audience. Optional when updating an audience. | 48.0              |
Attach a top-level title: ConnectApi Input Classes

**ConnectApi.AuditParamsRequest**

Audit Parameters input.

This class is abstract.

Superclass of ConnectApi.BaseRequest.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>email</td>
<td>String</td>
<td>Email of the client that made the request.</td>
<td>Optional</td>
<td>50.0</td>
</tr>
<tr>
<td>ipAddress</td>
<td>String</td>
<td>IP address of the client that made the request.</td>
<td>Optional</td>
<td>50.0</td>
</tr>
<tr>
<td>macAddress</td>
<td>String</td>
<td>Mac address of the client that made the request.</td>
<td>Optional</td>
<td>50.0</td>
</tr>
<tr>
<td>phone</td>
<td>String</td>
<td>Phone number of the client that made the request.</td>
<td>Optional</td>
<td>50.0</td>
</tr>
</tbody>
</table>

**ConnectApi.AuthApiPaymentMethodRequest**

Payment method input representation for payment authorizations.

Subclass of ConnectApi.BaseApiPaymentMethodRequest.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>card</td>
<td>ConnectApi.Card PaymentMethodRequest</td>
<td>Card payment method information.</td>
<td>Required</td>
<td>51.0</td>
</tr>
</tbody>
</table>

**ConnectApi.AuthorizationReversalRequest**

Authorization reversal input consumed by authorization reversal service.

Subclass of ConnectApi.BaseRequest.
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>accountId</td>
<td>String</td>
<td>Account for the payment authorization reversal. Must match the payment authorization's account.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
<tr>
<td>amount</td>
<td>Double</td>
<td>Amount of adjustment applied to the payment authorization.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>comments</td>
<td>String</td>
<td>Users can add comments to provide additional details about a record. Maximum of 1,000 characters.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
<tr>
<td>effectiveDate</td>
<td>Datetime</td>
<td>Date that the adjustment takes effect on the authorization.</td>
<td>Required</td>
<td>51.0</td>
</tr>
</tbody>
</table>

**ConnectApi.AuthorizationRequest**

Payment Authorization input consumed by Payment Authorization service.

Subclass of `ConnectApi.BaseRequest`.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>accountId</td>
<td>String</td>
<td>Salesforce account that contains the payment transaction being authorized.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>amount</td>
<td>Double</td>
<td>Authorization amount.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>comments</td>
<td>String</td>
<td>Optional comments for the payment authorization.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Three-letter ISO 4217 currency code associated with the payment group record.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>effectiveDate</td>
<td>Datetime</td>
<td>Date that the authorization will be applied to the transaction.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>paymentGatewayId</td>
<td>String</td>
<td>Payment gateway that processes the authorization.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>paymentGroup</td>
<td><code>ConnectApi.PaymentGroupRequest</code></td>
<td>Payment group for the authorization. The payload must reference either a paymentGroup or a paymentGroupId, but not both.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
<tr>
<td>paymentMethod</td>
<td><code>ConnectApi.AuthApiPaymentMethodRequest</code></td>
<td>Payment method used in the payment gateway for the authorization transaction.</td>
<td>Required</td>
<td>51.0</td>
</tr>
</tbody>
</table>
**ConnectApi.BannerPhotoInput**

A banner photo.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>cropHeight</td>
<td>Integer</td>
<td>Height of the crop rectangle in pixels.</td>
<td>Optional</td>
<td>36.0</td>
</tr>
<tr>
<td>cropWidth</td>
<td>Integer</td>
<td>Width of the crop rectangle in pixels.</td>
<td>Optional</td>
<td>36.0</td>
</tr>
<tr>
<td>cropX</td>
<td>Integer</td>
<td>X position of the crop rectangle from the left edge of the image in pixels.</td>
<td>Optional</td>
<td>36.0</td>
</tr>
<tr>
<td>cropY</td>
<td>Integer</td>
<td>Y position of the crop rectangle from the top edge of the image in pixels.</td>
<td>Optional</td>
<td>36.0</td>
</tr>
<tr>
<td>fileId</td>
<td>String</td>
<td>18 character ID of an existing file. The key prefix must be 069 and the file must be an image and be smaller than 2 GB.</td>
<td>Required</td>
<td>36.0</td>
</tr>
</tbody>
</table>

**Note:** Images uploaded on the Group page and on the User page don’t have file IDs and therefore can’t be used.

| versionNumber | Integer | Version number of an existing file. If not provided, the latest version is used. | Optional | 36.0 |

**ConnectApi.BaseRequest**

Base parameters for making a request to the payment gateway.

This class is abstract.

Subclass of **ConnectApi.AuditParamsRequest**.

Superclass of:

- ConnectApi.AuthorizationRequest
- ConnectApi.AuthorizationReversalRequest
- ConnectApi.CaptureRequest
- ConnectApi.PaymentMethodTokenizationRequest
- ConnectApi.PostAuthRequest
- ConnectApi.RefundRequest
- ConnectApi.SaleRequest
### ConnectApi.BaseApiPaymentMethodRequest

Payment method API input representation.

This class is abstract.

Superclass of:
- ConnectApi.AuthApiPaymentMethodRequest
- ConnectApi.PostAuthApiPaymentMethodRequest
- ConnectApi.SaleApiPaymentMethodRequest

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>address</td>
<td>ConnectApi.AddressRequest</td>
<td>Payment method address.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>Payment method record ID. Used in payment transactions.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>saveForFuture</td>
<td>Boolean</td>
<td>Shows whether Salesforce saves the payment method for future use.</td>
<td>Required</td>
<td>51.0</td>
</tr>
</tbody>
</table>

### ConnectApi.BasePaymentMethodRequest

Base payment method input representation.

This class is abstract.

Superclass of:
- ConnectApi.AlternativePaymentMethod
- ConnectApi.CardPaymentMethodRequest

No additional properties.

### ConnectApi.BatchInput

Construct a set of inputs to be passed into a method at the same time.

Use this constructor when there isn’t a binary input:

```java
ConnectApi.BatchInput(Object input)
```
Use this constructor to pass one binary input:

```java
ConnectApi.BatchInput(Object input, ConnectApi.BinaryInput binary)
```

Use this constructor to pass multiple binary inputs:

```java
ConnectApi.BatchInput(Object input, List<ConnectApi.BinaryInput> binaries)
```

The constructors takes these parameters:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>input</td>
<td>Object</td>
<td>An individual input object to be used in the batch operation. For example, for <code>postFeedElementBatch()</code>, this should be <code>ConnectApi.FeedElementInput</code>.</td>
<td>32.0</td>
</tr>
<tr>
<td>binary</td>
<td><code>ConnectApi.BinaryInput</code></td>
<td>A binary file to associate with the input object.</td>
<td>32.0</td>
</tr>
<tr>
<td>binaries</td>
<td><code>List&lt;ConnectApi.BinaryInput&gt;</code></td>
<td>A list of binary files to associate with the input object.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

- Post a Batch of Feed Elements
- Post a Batch of Feed Elements with a New (Binary) File

**ConnectApi.BinaryInput**

Create a `ConnectApi.BinaryInput` object to attach files to feed items and comments and to add repository files.

The constructor is:

```java
ConnectApi.BinaryInput(blob, contentType, filename)
```

The constructor takes these arguments:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>blob</td>
<td>Blob</td>
<td>Contents of the file to be used for input</td>
<td>28.0</td>
</tr>
<tr>
<td>contentType</td>
<td>String</td>
<td>MIME type description of the content, such as <code>image/jpg</code></td>
<td>28.0</td>
</tr>
<tr>
<td>filename</td>
<td>String</td>
<td>File name with the file extension, such as <code>UserPhoto.jpg</code></td>
<td>28.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

- Post a Feed Element with a New File (Binary) Attachment
- Post a Comment with a New File
- `ConnectApi.BatchInput`
ConnectApi.BookmarksCapabilityInput
Create or update a bookmark on a feed element.
This class is a subclass of ConnectApi.FeedElementCapabilityInput.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isBookmarked ByCurrentUser</td>
<td>Boolean</td>
<td>Specifies if the feed element should be bookmarked for the user (true) or not (false).</td>
<td>No</td>
<td>32.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.FeedElementCapabilitiesInput

ConnectApi.BotVersionActivationInput
Activation status of the bot version.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>status</td>
<td>ConnectApi.BotVersionActivationStatus</td>
<td>Activation status of the bot version. Values are:</td>
<td>Optional</td>
<td>50.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Active</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Inactive</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Activation status must be specified in the status or postBody parameter.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

ConnectApi.CanvasCapabilityInput
Create or update a canvas app associated with a feed element.
This class is a subclass of ConnectApi.FeedElementCapabilityInput.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>String</td>
<td>A description of the canvas app. The maximum size is 255 characters.</td>
<td>Optional</td>
<td>32.0</td>
</tr>
<tr>
<td>developerName</td>
<td>String</td>
<td>The API name (developer name) of the connected app.</td>
<td>Required</td>
<td>32.0</td>
</tr>
<tr>
<td>height</td>
<td>String</td>
<td>The height of the canvas app in pixels.</td>
<td>Optional</td>
<td>32.0</td>
</tr>
<tr>
<td>namespacePrefix</td>
<td>String</td>
<td>A unique namespace prefix for the canvas app.</td>
<td>Optional</td>
<td>32.0</td>
</tr>
<tr>
<td>parameters</td>
<td>String</td>
<td>JSON parameters passed to the canvas app.</td>
<td>Optional</td>
<td>32.0</td>
</tr>
</tbody>
</table>
### ConnectApi.CapacityRequestInputRepresentation

Request related to a location’s fulfillment order capacity.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>actionRequestId</td>
<td>String</td>
<td>Unique string that identifies the request. Can be a UUID. Use the action request IDs in response data to identify which requests succeeded or failed.</td>
<td>Required</td>
<td>55.0</td>
</tr>
<tr>
<td>locationId</td>
<td>String</td>
<td>ID of the location associated with the request.</td>
<td>Required</td>
<td>55.0</td>
</tr>
</tbody>
</table>

### ConnectApi.CaptureRequest

Payment capture input consumed by the payment capture service. Subclass of ConnectApi.BaseRequest.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>accountId</td>
<td>String</td>
<td>ID of the account linked to the capture request.</td>
<td>Optional</td>
<td>50.0</td>
</tr>
<tr>
<td>amount</td>
<td>Double</td>
<td>Amount captured from the previous authorization.</td>
<td>Required</td>
<td>50.0</td>
</tr>
<tr>
<td>clientContext</td>
<td>String</td>
<td>Context for payment APIs. Used for a payment caller to re-establish context.</td>
<td>Optional</td>
<td>50.0</td>
</tr>
<tr>
<td>comments</td>
<td>String</td>
<td>Comments for the payment capture.</td>
<td>Optional</td>
<td>50.0</td>
</tr>
<tr>
<td>effectiveDate</td>
<td>Datetime</td>
<td>Date when the payment becomes effective.</td>
<td>Optional</td>
<td>50.0</td>
</tr>
<tr>
<td>paymentGroup</td>
<td>ConnectApi. PaymentGroupRequest</td>
<td>Details about the payment group record associated with the payment request.</td>
<td>Optional</td>
<td>50.0</td>
</tr>
</tbody>
</table>
ConnectApi.AlternativePaymentMethod

A payment method that doesn’t have a defined Salesforce entity such as CardPaymentMethod or DigitalWallet. Common examples of alternative payment methods include CashOnDeliver, Klarna, and Direct Debit. AlternativePaymentMethod functions the same as any other type of payment method for processing transactions in the payment gateway.

Subclass of ConnectApi.BasePaymentMethodRequest

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>accountId</td>
<td>String</td>
<td>Salesforce Payments account to which this payment method is linked.</td>
<td>Required</td>
<td>54.0</td>
</tr>
<tr>
<td>comments</td>
<td>String</td>
<td>Details about a record added by a user. Maximum of 1,000 characters.</td>
<td>Optional</td>
<td>54.0</td>
</tr>
<tr>
<td>email</td>
<td>String</td>
<td>Email address of the card holder.</td>
<td>Optional</td>
<td>54.0</td>
</tr>
<tr>
<td>gatewayToken</td>
<td>String</td>
<td>A unique, alphanumeric ID, called a token, that a payment gateway generates when it first processes a payment. The token replaces the actual payment data so that the data is kept secure. This token is stored as encrypted text, and can be used for recurring payments.</td>
<td>Required</td>
<td>54.0</td>
</tr>
<tr>
<td>gatewayToken Details</td>
<td>String</td>
<td>Detailed information about the gateway token.</td>
<td>Required</td>
<td>54.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name that you assign to the payment method object.</td>
<td>Optional</td>
<td>54.0</td>
</tr>
</tbody>
</table>

ConnectApi.CardPaymentMethodRequest

Card payment method input representation.

Subclass of ConnectApi.BasePaymentMethodRequest.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>accountId</td>
<td>String</td>
<td>Salesforce Payments account to which this payment method is linked.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>autoPay</td>
<td>Boolean</td>
<td>Indicates whether a token for recurring payments is being requested (true) or not (false). The token lets the payment method be used for recurring payments.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>cardCategory</td>
<td>ConnectApi.CardCategory</td>
<td>Card processing type. Valid values are: • CreditCard • DebitCard</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>Property</td>
<td>Type</td>
<td>Description</td>
<td>Required or Optional</td>
<td>Available Version</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------</td>
<td>--------------------------------------------------</td>
<td>-----------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>cardHolder</td>
<td>String</td>
<td>First name of the card holder.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>FirstName</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cardHolder</td>
<td>String</td>
<td>Last name of the card holder.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>LastName</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cardHolderName</td>
<td>String</td>
<td>Full name of the card holder.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>cardNumber</td>
<td>String</td>
<td>Card number.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>cardType</td>
<td>String</td>
<td>Card network type. Valid values are: AmericanExpress, DinersClub, JCB, MasterCard, Maestro, Visa</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>comments</td>
<td>String</td>
<td>Optional comments for the card payment method.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
<tr>
<td>cvv</td>
<td>String</td>
<td>Card Verification Value.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>email</td>
<td>String</td>
<td>Email address of the card holder.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>expiryMonth</td>
<td>Integer</td>
<td>Card expiration month.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>expiryYear</td>
<td>Integer</td>
<td>Card expiration year.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>nickName</td>
<td>String</td>
<td>Optional nickname for the card.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
<tr>
<td>startMonth</td>
<td>Integer</td>
<td>Month the card becomes active.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
<tr>
<td>startYear</td>
<td>Integer</td>
<td>Year the card becomes active.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
</tbody>
</table>

**ConnectApi.CartInput**

A cart.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Currency ISO code of the cart.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>effectiveAccountId</td>
<td>String</td>
<td>ID of the buyer account or guest buyer profile for which the request is made.</td>
<td>Optional</td>
<td>49.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If unspecified, the default value is determined from context.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Property</td>
<td>Type</td>
<td>Description</td>
<td>Required or Optional</td>
<td>Available Version</td>
</tr>
<tr>
<td>------------------</td>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>isSecondary</td>
<td>Boolean</td>
<td>Specifies whether the cart is secondary (true) or not (false). If unspecified, defaults to false.</td>
<td>Optional</td>
<td>53.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the cart. The name can have up to 250 Unicode characters. If unspecified, defaults to the generated name.</td>
<td>Optional</td>
<td>49.0</td>
</tr>
<tr>
<td>orderOwnerId</td>
<td>String</td>
<td>ID of the owner of the order.</td>
<td>Optional</td>
<td>58.0</td>
</tr>
<tr>
<td>type</td>
<td>ConnectApi.CartType</td>
<td>Type of cart. Vaues are:</td>
<td>Optional</td>
<td>49.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Cart—Cart created by a customer.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ReadOnly—Clone of a Template cart that the customer can check out with.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Template—Cart created by an internal user.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>If unspecified, defaults to Cart.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>typeAsString</td>
<td>String</td>
<td>Type of the cart provided as a string.</td>
<td>Optional</td>
<td>59.0</td>
</tr>
</tbody>
</table>

ConnectApi.cartCouponInput

Cart coupon input.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>couponCode</td>
<td>String</td>
<td>The coupon code.</td>
<td>Required</td>
<td>54.0</td>
</tr>
</tbody>
</table>

ConnectApi.CartInventoryReservationInputRepresentation (Pilot)

Input representation to create or update a reservation.

**Note:** This feature is not generally available and is being piloted with certain Customers subject to additional terms and conditions. It is not part of your purchased Services. This feature is subject to change, may be discontinued with no notice at any time in Salesforce's sole discretion, and Salesforce may never make this feature generally available. Make your purchase decisions only on the basis of generally available products and features. This feature is made available on an AS IS basis and use of this feature is at your sole risk.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>durationInSeconds</td>
<td>Integer</td>
<td>Reservation duration in seconds.</td>
<td>Required</td>
<td>58.0</td>
</tr>
</tbody>
</table>
## ConnectApi.CartItemInput

An item in a cart.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>cartDeliveryGroupId</td>
<td>String</td>
<td>ID of the cart delivery group.</td>
<td>Optional</td>
<td>59.0</td>
</tr>
<tr>
<td>productId</td>
<td>String</td>
<td>ID of the product.</td>
<td>Required when adding an item to a cart</td>
<td>49.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Not supported when updating a cart item</td>
<td></td>
</tr>
<tr>
<td>quantity</td>
<td>String</td>
<td>Quantity of the cart item. Use a value that can be converted to BigDecimal.</td>
<td>Required</td>
<td>49.0</td>
</tr>
</tbody>
</table>
| type                 | ConnectApi.CartItemType | Type of item in a cart. Value is Product.  
- DeliveryCharge  
- Product | Required when adding an item to a cart | 49.0              |
|                      |          |                                                                             | Not supported when updating a cart item |                  |

## ConnectApi.CartItemPromotionCollectionInputRepresentation

Promotions for a cart item.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>items</td>
<td>List&lt;ConnectApi.CartItemPromotionInputRepresentation&gt;</td>
<td>List of cart items to get the associated promotions.</td>
<td>Required</td>
<td>52.0</td>
</tr>
</tbody>
</table>

## ConnectApi.CartItemPromotionInputRepresentation

ID of a cart item associated with a promotion.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>cartItemId</td>
<td>String</td>
<td>ID of the item associated with the cart.</td>
<td>Optional</td>
<td>52.0</td>
</tr>
</tbody>
</table>

## ConnectApi.CartMessagesVisibilityInput

Set the visibility for cart messages.
### ConnectApi.CartToWishlistInput

Copy products from a cart to a wishlist.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>wishlistId</td>
<td>String</td>
<td>ID of the wishlist to copy cart products to.</td>
<td>Required</td>
<td>50.0</td>
</tr>
</tbody>
</table>

### ConnectApi.CdpCalculatedInsightInput

Input representation for a calculated insight.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>apiName</td>
<td>String</td>
<td>API name of the calculated insight with suffix __cio.</td>
<td>Required for creating a calculated insight</td>
<td>57.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Optional for updating a calculated insight</td>
<td></td>
</tr>
<tr>
<td>createdFrom Package</td>
<td>Boolean</td>
<td>Specifies whether the calculated insight was created from an installed package (true) or not (false).</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>dataSpaceName</td>
<td>String</td>
<td>Name of the data space.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
</tbody>
</table>
| definitionType            | ConnectApi.CalculatedInsightDefinitionTypeEnum | Definition type of the calculated insight. Values are: 
- CalculatedMetric 
- ExternalMetric 
- StreamingMetric | Required for creating a calculated insight | 57.0              |
|                          |                     |                                                                             | Optional for updating a calculated insight |                  |
| description               | String              | Calculated insight description.                                            | Optional             | 57.0              |
| displayName               | String              | Calculated insight display name.                                           | Required for creating a calculated insight | 57.0              |
|                          |                     |                                                                             | Optional for updating a calculated insight |                  |
### Available Version

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>draft</td>
<td>Boolean</td>
<td>Specifies whether to save the calculated insight as draft (<strong>true</strong>) or not (<strong>false</strong>).</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>expression</td>
<td>String</td>
<td>Calculated insight ANSI SQL expression.</td>
<td>Required for creating a calculated insight</td>
<td>57.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Optional for updating a calculated insight</td>
<td></td>
</tr>
</tbody>
</table>

### ConnectApi.CdpIdentityResolutionConfigInput

Input representation for creating an identity resolution ruleset.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>configurationType</td>
<td>ConnectApi.CdpIdentityResolutionConfigurationType</td>
<td>Source object for an identity resolution ruleset. Values are: • Account • Individual</td>
<td>Required</td>
<td>57.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the identity resolution ruleset.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>doesRunAutomatically</td>
<td>Boolean</td>
<td>Specifies whether automatic job run scheduling is enabled for the ruleset (<strong>true</strong>) or not (<strong>false</strong>). If unspecified, defaults to <strong>false</strong>.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>label</td>
<td>String</td>
<td>User friendly name of the identity resolution ruleset.</td>
<td>Required</td>
<td>57.0</td>
</tr>
<tr>
<td>rulesetId</td>
<td>String</td>
<td>Extended ID of the ruleset used to differentiate between rulesets created for comparison. The ruleset ID must be unique and can't be longer than 4 characters.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
</tbody>
</table>
**ConnectApi.CdpIdentityResolutionConfigPatchInput**

Input representation for updating an identity resolution ruleset.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the identity resolution ruleset.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>doesRunAutomatically</td>
<td>Boolean</td>
<td>Specifies whether automatic job run scheduling is enabled for the ruleset (true) or not (false). If unspecified, defaults to false.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>label</td>
<td>String</td>
<td>User friendly name of the identity resolution ruleset.</td>
<td>Required</td>
<td>57.0</td>
</tr>
<tr>
<td>matchRules</td>
<td>List&lt;String&gt;</td>
<td>List of match rules for the identity resolution ruleset.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>reconciliationRules</td>
<td>List&lt;String&gt;</td>
<td>List of reconciliation rules for the identity resolution ruleset.</td>
<td>Required</td>
<td>57.0</td>
</tr>
</tbody>
</table>

**ConnectApi.CdpIdentityResolutionMatchCriterion**

Input representation for an identity resolution ruleset's match rule criterion.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>caseSensitiveMatch</td>
<td>Boolean</td>
<td>Specifies whether the criterion match is case sensitive (true) or not (false). Available only when matching is based on the party identifier.</td>
<td>Optional</td>
<td>58.0</td>
</tr>
<tr>
<td>entityName</td>
<td>String</td>
<td>API name of the Data Model Object the match rule applies to.</td>
<td>Required</td>
<td>57.0</td>
</tr>
<tr>
<td>fieldName</td>
<td>String</td>
<td>Name of the field the criterion applies to.</td>
<td>Required</td>
<td>57.0</td>
</tr>
<tr>
<td>matchMethodType</td>
<td>String</td>
<td>Match method for a match rule criterion. Values are:</td>
<td>Required</td>
<td>57.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Exact—Exact match.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- ExactNormalized—Exact normalized match.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Fuzzy—Fuzzy match with medium precision.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- FuzzyHigh—Fuzzy match with high precision.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Property</td>
<td>Type</td>
<td>Description</td>
<td>Required or Optional</td>
<td>Available Version</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>---------------------------</td>
<td>-------------------------------------------------------------------------------</td>
<td>----------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>partyIdentificationInfo</td>
<td>ConnectApi.CdpIdentityResolutionMatchCriterionPartyIdentificationInfo</td>
<td>Party Identifier information.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>shouldMatchOnBlank</td>
<td>Boolean</td>
<td>Specifies whether blank fields can be used for matching (true) or not (false).</td>
<td>Required</td>
<td>57.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
ConnectApi.CdpIdentityResolutionMatchRule

**ConnectApi.CdpIdentityResolutionMatchCriterionPartyIdentificationInfo**
Input representation for information when party identification is used in an identity resolution ruleset's match rule criterion.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>partyName</td>
<td>String</td>
<td>Party identification name.</td>
<td>Required if the match rule criterion uses party identification for matching</td>
<td>57.0</td>
</tr>
<tr>
<td>partyType</td>
<td>String</td>
<td>Party identification type.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
ConnectApi.CdpIdentityResolutionMatchCriterion

**ConnectApi.CdpIdentityResolutionMatchRule**
Input representation for an identity resolution ruleset's match rule.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>criteria</td>
<td>List&lt;ConnectApi.CdpIdentityResolutionMatchCriterion&gt;</td>
<td>Object and field the match rule applies to and the match method applied.</td>
<td>Required</td>
<td>57.0</td>
</tr>
</tbody>
</table>
### ConnectApi.CdpIdentityResolutionReconciliationFieldRule

Input representation for an identity resolution ruleset’s reconciliation rule for a field.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>fieldName</td>
<td>String</td>
<td>The field that this reconciliation rule applies to.</td>
</tr>
</tbody>
</table>
| ruleType          | ConnectApi.CdpIdentityResolutionReconciliationRuleType | Default reconciliation rule applied to fields in the object the reconciliation rule applies to. Values are:
|                   |                               | - LastUpdated                                                               |
|                   |                               | - MostFrequent                                                              |
|                   |                               | - SourceSequence                                                            |
| shouldIgnoreEmptyValue | Boolean                     | Specifies whether to ignore an empty value (true) or not (false).          |
| sources           | List<ConnectApi.CdpIdentityResolutionReconciliationSource> | If ruleType is SourceSequence, a prioritized list of data sources.          |

**Available Version**: Required or Optional

#### ConnectApi.CdpIdentityResolutionReconciliationRule

Input representation for an identity resolution ruleset’s default reconciliation rule for an object.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>entityName</td>
<td>String</td>
<td>API name of the Data Model Object the reconciliation rule applies to.</td>
</tr>
</tbody>
</table>

**Available Version**: Required or Optional
**Available Version**

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>fields</td>
<td>List&lt;ConnectApi.CdpIdentityResolutionReconciliationFieldRule&gt;</td>
<td>Field-specific reconciliation rules that override this default rule for the specified field.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
</tbody>
</table>
| ruleType                  | ConnectApi.CdpIdentityResolutionReconciliationRuleType | Default reconciliation rule applied to fields in the object the reconciliation rule applies to. Values are:  
  - LastUpdated  
  - MostFrequent  
  - SourceSequence | Required             | 57.0              |
| shouldIgnoreEmptyValue    | Boolean                                | Specifies whether to ignore an empty value (true) or not (false).            | Required             | 57.0              |
| sources                   | List<ConnectApi.CdpIdentityResolutionReconciliationSource> | If ruleType is SourceSequence, a list of data sources in priority order.      | Required if ruleType is SourceSequence | 57.0              |

**SEE ALSO:**
- ConnectApi.CdpIdentityResolutionConfigInput  
- ConnectApi.CdpIdentityResolutionConfigPatchInput

**ConnectApi.CdpIdentityResolutionReconciliationSource**

Input representation for an identity resolution default reconciliation rule or field-specific rule using the SourceSequence match method.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>If the ruleType for a reconciliation rule is SourceSequence, API name of a source Data Lake Object.</td>
<td>Required if ruleType is SourceSequence</td>
<td>57.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.CdpIdentityResolutionReconciliationRule  
- ConnectApi.CdpIdentityResolutionReconciliationFieldRule  
- ConnectApi.CdpIdentityResolutionReconciliationFieldRule

**ConnectApi.CdpIdentityResolutionRunNowInput**

Input representation for running an identity resolution ruleset job on demand.
### ConnectApi.CdpQueryInput

Data query input.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>sql</td>
<td>String</td>
<td>ANSI-standard SQL query.</td>
<td>Required</td>
<td>52.0</td>
</tr>
</tbody>
</table>

### ConnectApi.CdpSegmentDbtInput

Segment dbt input.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>models</td>
<td>List&lt;ConnectApi.CdpSegmentDbtModelInput&gt;</td>
<td>List of models. The segment data build tool currently supports a single SQL model.</td>
<td>Required</td>
<td>55.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

- ConnectApi.CdpSegmentInput

### ConnectApi.CdpSegmentDbtModelInput

Segment dbt model input.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>Dbt model name.</td>
<td>Required</td>
<td>55.0</td>
</tr>
</tbody>
</table>
### ConnectApi.CdpSegmentDbtInput

Segment input.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>sql</td>
<td>String</td>
<td>Dbt SQL.</td>
<td>Required</td>
<td>55.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Dbt SQL date strings must be in ISO 8601 format, for example, 2011-02-25T18:24:31.000Z.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>For details about supported validations, see <a href="#">Supported Validations for Segment Data Build Tool Model SQL</a>.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

See also:
- ConnectApi.CdpSegmentDbtInput

### ConnectApi.CdpSegmentInput

Segment input.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>additionalMetadata</td>
<td>Map&lt;String, String&gt;</td>
<td>Map of additional metadata.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>dataSpace</td>
<td>String</td>
<td>Segment dataspace. In API version 59.0 and later, this property is not available. Use the dataspace request parameter instead.</td>
<td>Optional</td>
<td>57.0–58.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Segment description.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>developerName</td>
<td>String</td>
<td>Segment developer name.</td>
<td>Required</td>
<td>55.0</td>
</tr>
<tr>
<td>displayName</td>
<td>String</td>
<td>Segment display name.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>includeDbt</td>
<td>ConnectApi.CdpSegmentDbtInput</td>
<td>Segment data build tool.</td>
<td>Required</td>
<td>55.0</td>
</tr>
<tr>
<td>publishSchedule</td>
<td>ConnectApi.PublishSchedule</td>
<td>Publish refresh schedule. Values are: one—Refreshes every hour. Used to rapidly publish UI and DBT-based segments.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
</tbody>
</table>
Available Version | Required or Optional
--- | ---
Four—Refreshes every four hours. Used to rapidly publish UI and DBT-based segments. | Optional if publishSchedule isn’t specified
Twelve—Refreshes every twelve hours. |
TwentyFour—Refreshes every twenty-four hours. |

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>publishScheduleEndDate</td>
<td>String</td>
<td>Date indicating the end of the publish schedule.</td>
<td>Optional if publishSchedule isn’t specified</td>
<td>55.0</td>
</tr>
<tr>
<td>publishScheduleStartDateTime</td>
<td>String</td>
<td>Datetime indicating the start of the publish schedule.</td>
<td>Optional if publishSchedule isn’t specified</td>
<td>55.0</td>
</tr>
<tr>
<td>segmentOnApiName</td>
<td>String</td>
<td>API name of the SegmentOn entity.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
</tbody>
</table>
| segmentType | ConnectApi.SegmentType | Type of segment. Value is:  
- Dbt—Data build tool  
After a segment is created, the segment type can’t be changed. | Required for creating a segment  
Not supported for updating a segment | 55.0 |

**ConnectApi.ChangeInputRepresentation**

A list of changes to OrderItemSummaries that make up an order change, such as a cancel or return.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>changeItems</td>
<td>List&lt;ConnectApi.ChangeItemInputRepresentation&gt;</td>
<td>List of changes to OrderItemSummaries.</td>
<td>Required</td>
<td>48.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

`previewCancel(orderSummaryId, changelInput)`  
`previewReturn(orderSummaryId, changelInput)`  
`submitCancel(orderSummaryId, changelInput)`  
`submitReturn(orderSummaryId, changelInput)`

**ConnectApi.ChangeItemInputRepresentation**

Change to an OrderItemSummary, such as a return or cancel. You specify whether to prorate the associated shipping charge based on the price change. The OrderItemSummary can’t be a shipping charge.
### Change Item Fee Input Representation

Input representation for Change Item Fee Input

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>amount</td>
<td>Double</td>
<td>Positive value used to calculate the fee amount, as described by the amountType.</td>
<td>Required</td>
<td>57.0</td>
</tr>
<tr>
<td>amountType</td>
<td>String</td>
<td>Describes how the fee is calculated. Valid values are:</td>
<td>Required</td>
<td>57.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AmountWithTax—Value of amount. is the fee amount, including tax.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AmountWithoutTax—Value of amount. is the fee amount, not including tax. Tax is calculated on the value and added.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Description

- **Percentage**—Value of amount is a percentage. To determine the fee amount, `amount` is divided by 100, and then multiplied by the TotalPrice and TotalTaxAmount of the associated OrderItemSummary, prorated for the quantity being returned.

- **PercentageGross**—Value of amount is a percentage. To determine the fee amount, `amount` is divided by 100, and then multiplied by the TotalLineAmountWithTax of the associated OrderItemSummary, prorated for the quantity being returned.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the fee.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>priceBookEntryId</td>
<td>String</td>
<td>ID of the price book entry associated with the fee product.</td>
<td>Required unless price books are optional in the org</td>
<td>57.0</td>
</tr>
<tr>
<td>product2Id</td>
<td>String</td>
<td>ID of the product representing the fee.</td>
<td>Required</td>
<td>57.0</td>
</tr>
<tr>
<td>reason</td>
<td>String</td>
<td>Reason for the fee. The value must match an entry in the OrderProductSummaryChange object’s Reason picklist.</td>
<td>Required</td>
<td>57.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

- ConnectApi.ChangelInputRepresentation
- ConnectApi.ChangelItemInputRepresentation
- previewCancel(orderSummaryId, changelInput)
- previewReturn(orderSummaryId, changelInput)
- submitCancel(orderSummaryId, changelInput)
- submitReturn(orderSummaryId, changelInput)

### ConnectApi.ChatterGroupInput

Chatter group input.
### Announcement ID

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>announcement</td>
<td>String</td>
<td>The 18-character ID of an announcement. An announcement displays in a designated location in the Salesforce UI until 11:59 p.m. on its expiration date, unless it's deleted or replaced by another announcement.</td>
<td>31.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>canHaveChatterGuests</td>
<td>Boolean</td>
<td>true if this group allows Chatter customers, false otherwise. After this property is set to true, it cannot be set to false.</td>
<td>29.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>The &quot;Description&quot; section of the group.</td>
<td>29.0</td>
</tr>
<tr>
<td>information</td>
<td>ConnectApi.GroupInformationInput</td>
<td>The &quot;Information&quot; section of a group. If the group is private, this section is visible only to members.</td>
<td>28.0</td>
</tr>
<tr>
<td>isArchived</td>
<td>Boolean</td>
<td>true if the group is archived, false otherwise. Defaults to false.</td>
<td>29.0</td>
</tr>
<tr>
<td>isAutoArchiveDisabled</td>
<td>Boolean</td>
<td>true if automatic archiving is turned off for the group, false otherwise. Defaults to false.</td>
<td>29.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>The name of the group.</td>
<td>29.0</td>
</tr>
<tr>
<td>owner</td>
<td>String</td>
<td>The ID of the group owner. This property is available for PATCH requests only.</td>
<td>29.0</td>
</tr>
<tr>
<td>visibility</td>
<td>ConnectApi.GroupVisibilityType</td>
<td>Group visibility type.</td>
<td>29.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- PrivateAccess—Only members of the group can see posts to this group.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- PublicAccess—All users within the Experience Cloud site can see posts to this group.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Unlisted—Reserved for future use.</td>
<td></td>
</tr>
</tbody>
</table>

See also:
- `createGroup(communityId, groupInput)`
- `updateGroup(communityId, groupId, groupInput)`

### ConnectApi.ChatterStreamInput

A Chatter feed stream.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the stream, up to 1,000 characters.</td>
<td>Optional</td>
<td>39.0</td>
</tr>
<tr>
<td>Property</td>
<td>Type</td>
<td>Description</td>
<td>Required or Optional</td>
<td>Available Version</td>
</tr>
<tr>
<td>-----------------</td>
<td>--------------------------</td>
<td>----------------------------------------------------------------------------------------------------------</td>
<td>-----------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the stream, up to 120 characters.</td>
<td>Required when creating a stream</td>
<td>39.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Optional when updating a stream</td>
<td></td>
</tr>
<tr>
<td>subscriptions</td>
<td>List&lt;ConnectApi, Stream</td>
<td>List of up to 25 entities whose feeds are included in the stream.</td>
<td>Optional</td>
<td>39.0</td>
</tr>
<tr>
<td>ToAdd</td>
<td>Subscription Input&gt;</td>
<td>Adding an entity that is already added results in no operation. Including the same entity in subscriptionsToAdd and subscriptionsToRemove results in no operation.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>subscriptions</td>
<td>List&lt;ConnectApi, Stream</td>
<td>List of entities whose feeds are removed from the stream.</td>
<td>Optional when updating a stream</td>
<td>39.0</td>
</tr>
<tr>
<td>ToRemove</td>
<td>Subscription Input&gt;</td>
<td>Removing an entity that is already removed results in no operation. Including the same entity in subscriptionsToAdd and subscriptionsToRemove results in no operation.</td>
<td>Not supported when creating a stream</td>
<td></td>
</tr>
</tbody>
</table>

**ConnectApi.CommentCapabilitiesInput**

A container for all capabilities that can be included with a comment.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>content</td>
<td>ConnectApi.ContentCapabilityInput</td>
<td>Content to attach to the comment.</td>
<td>Optional</td>
<td>32.0</td>
</tr>
<tr>
<td>feedEntityShare</td>
<td>ConnectApi.FeedEntityShareCapabilityInput</td>
<td>Feed entity to share to the comment.</td>
<td>Optional</td>
<td>42.0</td>
</tr>
<tr>
<td>record</td>
<td>ConnectApi.RecordCapabilityInput</td>
<td>Existing knowledge article to attach to the comment.</td>
<td>Optional</td>
<td>42.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

ConnectApi.CommentInput

**ConnectApi.CommentInput**

Comment input used to add rich comments, for example, comments that include mentions or file attachments.
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>attachment</td>
<td>ConnectApi. FeedItem AttachmentInput</td>
<td>Specifies an attachment for the comment. Valid values are: • ContentAttachmentInput • NewFileAttachmentInput LinkAttachmentInput is not permitted for comments.</td>
<td>Optional</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>body</td>
<td>ConnectApi. MessageBodyInput</td>
<td>Description of message body. The body can contain up to 10,000 characters and 25 mentions. Because the character limit can change, clients should make a <code>describeSObjects()</code> call on the FeedItem or FeedComment object and look at the length of the <code>Body</code> or <code>CommentBody</code> field to determine the maximum number of allowed characters. To edit this property in a comment, use <code>updateComment(communityId, commentId, comment)</code>. Editing comments is supported in version 34.0 and later. Rich text and inline images are supported in comment bodies in version 35.0 and later. Entity links are supported in version 43.0 and later.</td>
<td>Required</td>
<td>28.0</td>
</tr>
<tr>
<td>capabilities</td>
<td>ConnectApi. CommentCapability Input</td>
<td>Specifies any capabilities for the comment, such as a file attachment.</td>
<td>Optional</td>
<td>32.0</td>
</tr>
<tr>
<td>threadParentId</td>
<td>String</td>
<td>ID of the parent comment for a threaded comment.</td>
<td>Optional</td>
<td>44.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- Post a Comment with a Mention
- Post a Comment with a New File
- Post a Comment with an Existing File
- Post a Rich-Text Comment with Inline Image
- Post a Rich-Text Feed Comment with a Code Block
- Edit a Comment
- `postCommentToFeedElement(communityId, feedElementId, comment, feedElementFileUpload)`
### ConnectApi.CommerceAddressInput

Commerce address input.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>addressType</td>
<td>String</td>
<td>Type of address, for example, Shipping or Billing.</td>
<td>Optional</td>
<td>54.0</td>
</tr>
<tr>
<td>city</td>
<td>String</td>
<td>The address city.</td>
<td>Optional</td>
<td>54.0</td>
</tr>
<tr>
<td>commerceAddressFieldInputList</td>
<td>List&lt;ConnectApi.CommerceAddressFieldInput&gt;</td>
<td>A list of custom address fields, if any.</td>
<td>Optional</td>
<td>54.0</td>
</tr>
<tr>
<td>companyName</td>
<td>String</td>
<td>The address company name.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>country</td>
<td>String</td>
<td>The address country.</td>
<td>Optional</td>
<td>54.0</td>
</tr>
<tr>
<td>countryCode</td>
<td>String</td>
<td>Two-character country code.</td>
<td>Optional</td>
<td>54.0–58.0</td>
</tr>
<tr>
<td>firstName</td>
<td>String</td>
<td>The address first name.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>isDefault</td>
<td>Boolean</td>
<td>Indicates whether a contact’s address is the preferred method of communication (true) or not (false). The default value is false.</td>
<td>Optional</td>
<td>54.0</td>
</tr>
<tr>
<td>lastName</td>
<td>String</td>
<td>The address last name.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>middleName</td>
<td>String</td>
<td>The address middle name.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the contact.</td>
<td>Required</td>
<td>54.0</td>
</tr>
<tr>
<td>phoneNumber</td>
<td>String</td>
<td>The address phone number.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>postalCode</td>
<td>String</td>
<td>Zip code or postal code for the address.</td>
<td>Optional</td>
<td>54.0</td>
</tr>
<tr>
<td>region</td>
<td>String</td>
<td>The address state.</td>
<td>Optional</td>
<td>54.0</td>
</tr>
<tr>
<td>regionCode</td>
<td>String</td>
<td>The address state code.</td>
<td>Optional</td>
<td>54.0–58.0</td>
</tr>
<tr>
<td>street</td>
<td>String</td>
<td>The address street.</td>
<td>Optional</td>
<td>54.0</td>
</tr>
</tbody>
</table>

### ConnectApi.CommerceAddressFieldInput

Commerce address field input. This is used to reference custom fields for the address.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>dataName</td>
<td>String</td>
<td>The name of the custom address field.</td>
<td>Required</td>
<td>54.0</td>
</tr>
<tr>
<td>text</td>
<td>String</td>
<td>The value of the custom address field.</td>
<td>Optional</td>
<td>54.0</td>
</tr>
</tbody>
</table>
### ConnectApi.ConfirmHeldFOCapacityInputRepresentation

Request to confirm held fulfillment order capacity at one or more locations. Can correspond to one action call.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>confirmHeldFOCapacityRequests</td>
<td>List&lt;ConnectApi.ConfirmHeldFOCapacityRequestInputRepresentation&gt;</td>
<td>List of requests to confirm held fulfillment order capacity at one or more locations.</td>
<td>Required</td>
<td>55.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ConfirmHeldFOCapacityRequestInputRepresentation

Request to confirm held fulfillment order capacity at one or more locations.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>allOrNothing</td>
<td>Boolean</td>
<td>Controls whether a single failed request cancels all other requests in the list (true) or whether some requests can succeed if others fail (false). The default value is false.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>capacityRequests</td>
<td>List&lt;ConnectApi.CapacityRequestInputRepresentation&gt;</td>
<td>List of requests to confirm held fulfillment order capacity. Each request is for one fulfillment order assigned to one location.</td>
<td>Required</td>
<td>55.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ContentCapabilityInput

Attach or update a file on a comment. Use this class to attach a new file or update a file that has already been uploaded to Salesforce.

This class is a subclass of ConnectApi.FeedElementCapabilityInput.

To attach or remove files from a feed post (instead of a comment) in version 36.0 and later, use ConnectApi.FilesCapabilityInput.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>contentDocumentId</td>
<td>String</td>
<td>ID of the existing content.</td>
<td>Required for existing content</td>
<td>32.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the file to be uploaded.</td>
<td>Optional</td>
<td>32.0</td>
</tr>
<tr>
<td>sharingOption</td>
<td>ConnectApi.FileSharingOption</td>
<td>Sharing option of the file. Values are:</td>
<td>Optional</td>
<td>35.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Allowed—Resharing of the file is allowed.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Restricted—Resharing of the file is restricted.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Property</td>
<td>Type</td>
<td>Description</td>
<td>Required or Optional</td>
<td>Available Version</td>
</tr>
<tr>
<td>----------</td>
<td>----------</td>
<td>-------------------------------------------------------------------------------------------------------</td>
<td>----------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>Title of the file. This value is used as the file name for new content. For example, if the title is My Title, and the file is a .txt file, the file name is My Title.txt.</td>
<td>Required for new content</td>
<td>32.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.FeedElementCapabilitiesInput

ConnectApi.ContentHubFieldValueInput

Fields of the item type.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the item field.</td>
<td>Required</td>
<td>39.0</td>
</tr>
<tr>
<td>value</td>
<td>String</td>
<td>Value of the item field.</td>
<td>Required</td>
<td>39.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.ContentHubItemInput

ConnectApi.ContentHubItemInput

Item type ID and fields of the item type.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>fields</td>
<td>List</td>
<td>List of fields for the item.</td>
<td>Required</td>
<td>39.0</td>
</tr>
<tr>
<td>itemTypeId</td>
<td>String</td>
<td>ID of the item type.</td>
<td>Required</td>
<td>39.0</td>
</tr>
</tbody>
</table>

ConnectApi.ContractInputRepresentation

Input to create and update contract.
### ConnectApi.CouponCodeRedemptionInput

Input representation for coupon code redemption.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>couponCodes</td>
<td>List&lt;String&gt;</td>
<td>Input representation for all coupon codes</td>
<td>Required</td>
<td>58.0</td>
</tr>
<tr>
<td>effectiveAccountId</td>
<td>String</td>
<td>ID of the account</td>
<td>Required</td>
<td>58.0</td>
</tr>
<tr>
<td>transactionId</td>
<td>String</td>
<td>ID of the transaction, which must be a valid cart ID</td>
<td>Required</td>
<td>58.0</td>
</tr>
</tbody>
</table>

### ConnectApi.CreateCreditMemoInputRepresentation

A list of change orders used to create a credit memo.
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>changeOrderIds</td>
<td>List&lt;String&gt;</td>
<td>List of IDs of the change orders.</td>
<td>Required</td>
<td>48.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
createCreditMemo(orderSummaryId, creditMemoInput)

**ConnectApi.CreateInvoiceFromChangeOrdersInputRepresentation**

OrderSummary and associated change orders to create Invoices for.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>changeOrderIds</td>
<td>List&lt;String&gt;</td>
<td>List of IDs of change orders to create Invoices for.</td>
<td>Required</td>
<td>56.0</td>
</tr>
<tr>
<td>orderSummaryId</td>
<td>String</td>
<td>ID of the associated Order Summary.</td>
<td>Required</td>
<td>56.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
createMultipleInvoices(invoicesInput)
ConnectApi.CreateMultipleInvoicesFromChangeOrdersInputRepresentation

**ConnectApi.CreateMultipleInvoicesFromChangeOrdersInputRepresentation**

Data about the change orders to create Invoices for.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>invoicesFrom</td>
<td>List&lt;ConnectApi.CreateInvoiceFromChangeOrdersInputRepresentation&gt;</td>
<td>List of OrderSummary IDs with the IDs of the associated change orders to create Invoices for. Each entry in the list generates one invoice, which combines the change orders in that entry.</td>
<td>Required</td>
<td>56.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
createMultipleInvoices(invoicesInput)

**ConnectApi.CreateOrderPaymentSummaryInputRepresentation**

An OrderSummary for which to create an OrderPaymentSummary, with the payment authorization or payments to include in it.
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>orderSummaryId</td>
<td>String</td>
<td>ID of the OrderSummary.</td>
<td>Required</td>
<td>48.0</td>
</tr>
<tr>
<td>paymentAuthorizationId</td>
<td>String</td>
<td>ID of the payment authorization.</td>
<td>Either a payment authorization or at least one payment is required.</td>
<td>48.0</td>
</tr>
<tr>
<td>paymentIds</td>
<td>List&lt;String&gt;</td>
<td>List of IDs of the payments.</td>
<td>Either a payment authorization or at least one payment is required.</td>
<td>48.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

`createOrderPaymentSummary(orderPaymentSummaryInput)`

## ConnectApi.CreateServiceAppointmentInput

Contains information to create a service appointment.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>assignedResources</td>
<td><a href="#">ConnectApi.AssignedResources</a></td>
<td>Represents the service resources to be assigned to a service appointment.</td>
<td>Optional</td>
<td>53.0</td>
</tr>
</tbody>
</table>

**Note:** When creating an appointment, use `extendedFields` to add values to any of the fields, including custom fields, in `assignedResources` as long as you have edit access to those fields.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>lead</td>
<td><a href="#">ConnectApi.LeadInput</a></td>
<td>Represents a prospect or lead.</td>
<td>Required if <code>serviceAppointment</code> isn't provided.</td>
<td>53.0</td>
</tr>
</tbody>
</table>

**Note:** Required to create a service appointment for unauthenticated guest users.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>schedulingPolicyId</td>
<td>String</td>
<td>The ID of the <code>AppointmentSchedulingPolicy</code> object. If no scheduling policy is passed in the request body, the default configurations are used. The only scheduling policy configuration that is used in determining time slots is the enforcement of account visiting hours.</td>
<td>Optional</td>
<td>53.0</td>
</tr>
</tbody>
</table>
### ServiceAppointment

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>serviceAppointment</td>
<td>ConnectApi.ServiceAppointmentInput</td>
<td>Represents the service appointment details to book an appointment.</td>
<td>Required if lead isn’t provided.</td>
<td>53.0</td>
</tr>
</tbody>
</table>

**Note:** When creating an appointment, use `extendedFields` to add values to any of the fields, including custom fields, in `assignedResources` as long as you have edit access to those fields.

### ConnectApi.CredentialInput

Credential input.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>authenticationProtocol</td>
<td>ConnectApi.Credential.AuthenticationProtocol</td>
<td>Authentication protocol of the external credential. Values are:</td>
<td>Required</td>
<td>56.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsSv4</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Custom</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Jwt</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• OAuth</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsSv4_STS—AWS Signature Version 4 with Security Token Service.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ClientCredentials.ClientSecret—OAuth 2.0 Client Credentials client secret.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ClientCredentialsJwtAssertion—OAuth 2.0 Client Credentials JSON Web Token assertion.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• JwtBearer—OAuth 2.0 JSON Web Token bearer flow.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NoAuthentication—No authentication.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• RolesAnywhere—AWS Signature Version 4 with Identity and Access Management (IAM) Roles Anywhere.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Application Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>credentials</td>
<td>Map&lt;String, ConnectApi.Credential CredentialValueInput&gt;</td>
<td>Map of protocol-specific credentials. Authentication protocols have credential allowlists and encryption rules.</td>
<td>Required</td>
<td>56.0</td>
</tr>
<tr>
<td>• AwsSv4</td>
<td></td>
<td>awsAccessKeyId (not encrypted), awsSecretAccessKey (encrypted), awsRoleArn (not encrypted)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Custom</td>
<td></td>
<td>Any credential name is valid (user sets encryption rules)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>externalCredential</td>
<td>String</td>
<td>Fully qualified developer name of the external credential.</td>
<td>Required</td>
<td>56.0</td>
</tr>
<tr>
<td>principalName</td>
<td>String</td>
<td>Name of the external credential named principal.</td>
<td>Required if principalType is NamedPrincipal</td>
<td>56.0</td>
</tr>
<tr>
<td>principalType</td>
<td>ConnectApi.CredentialPrincipalType</td>
<td>Type of credential principal. Values are:</td>
<td>Required</td>
<td>56.0</td>
</tr>
<tr>
<td>• AwsStsPrincipal</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• NamedPrincipal</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>• PerUserPrincipal</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### ConnectApi.CredentialCustomHeaderInput

Credential custom header input.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>headerName</td>
<td>String</td>
<td>Header name.</td>
<td>Required</td>
<td>58.0</td>
</tr>
<tr>
<td>headerValue</td>
<td>String</td>
<td>Header value.</td>
<td>Required</td>
<td>58.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>Header ID.</td>
<td>Optional</td>
<td>58.0</td>
</tr>
</tbody>
</table>
### ConnectApi.CredentialValueInput

Credential value input.

Authentication protocols have credential allowlists and encryption rules.

- **AwsSv4**—**awsAccessKeyId** (not encrypted), **awsSecretAccessKey** (encrypted), **awsRoleArn** (not encrypted)
- **Custom**—Any credential name is valid (user sets encryption rules)

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>encrypted</td>
<td>Boolean</td>
<td>Specifies whether the value of the credential is encrypted (<code>true</code>) or not (<code>false</code>).</td>
<td>Required</td>
<td>56.0</td>
</tr>
<tr>
<td>revision</td>
<td>Integer</td>
<td>Revision number of a short-lived credential, such as OAuthToken. If the provided revision isn’t the latest version, the authentication endpoint refreshes the credential.</td>
<td>Optional</td>
<td>58.0</td>
</tr>
<tr>
<td>value</td>
<td>String</td>
<td>Value of the credential.</td>
<td>Required</td>
<td>56.0</td>
</tr>
</tbody>
</table>

### ConnectApi.CustomListAudienceCriteriaInput

Criteria for the custom list type of custom recommendation audience.

Subclass of **ConnectApi.AudienceCriteriaInput**.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
</table>
| member OperationType | **ConnectApi.RecommendationAudienceMemberOperationType** | The operation to carry out on the audience members. Values are:  
- Add—Adds specified members to the audience.  
- Remove—Removes specified members from the audience. | Required to update a recommendation audience  
Don’t use or specify `null` to create a recommendation audience | 36.0              |
### ConnectApi.DirectMessageCapabilityInput

Create or update the members of a direct message.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>membersToAdd</td>
<td>List&lt;String&gt;</td>
<td>List of user IDs for members to include in the direct message.</td>
<td>Required when creating a direct message (POST)</td>
<td>39.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Optional when updating a direct message (PATCH)</td>
<td></td>
</tr>
<tr>
<td>membersToRemove</td>
<td>List&lt;String&gt;</td>
<td>List of user IDs for members to remove from the direct message.</td>
<td>Optional when updating a direct message (PATCH)</td>
<td>40.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Not supported when creating a direct message (POST)</td>
<td></td>
</tr>
<tr>
<td>subject</td>
<td>String</td>
<td>Subject of the direct message.</td>
<td>Optional when creating a direct message (POST)</td>
<td>39.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Not supported when updating a direct message (PATCH)</td>
<td></td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.FeedElementCapabilitiesInput
- ConnectApi.DistinctValueRefinementInput

### ConnectApi.DistinctValueRefinementInput

Attribute-based refinement with distinct values for product search.
This class is a subclass of ConnectApi.RefinementInput.
### ConnectApi.EnsureFundsAsyncInputRepresentation

ID of an Invoice to ensure funds for and apply them to.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>invoiceId</td>
<td>String</td>
<td>ID of the Invoice.</td>
<td>Required</td>
<td>48.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

`ensureFundsAsync(orderSummaryId, ensureFundsInput)`

### ConnectApi.EnsureRefundsAsyncInputRepresentation

ID of a credit memo to ensure refunds for, an amount of excess funds to refund, or both. At least one is required. Also includes any invoices for fees that reduce the refund amount, such as return fees. If multiple payment methods are available, you can specify how to distribute the refund.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>creditMemoId</td>
<td>String</td>
<td>ID of the credit memo that represents a refund amount.</td>
<td>Either</td>
<td>48.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>creditMemoId or excessFunds Amount is required</td>
<td></td>
<td></td>
</tr>
<tr>
<td>excessFunds</td>
<td>Double</td>
<td>Amount of excess funds to refund.</td>
<td>Either</td>
<td>49.0</td>
</tr>
<tr>
<td>Amount</td>
<td></td>
<td>excessFunds Amount or creditMemoId is required</td>
<td></td>
<td></td>
</tr>
<tr>
<td>invoicesToPay</td>
<td>List&lt;String&gt;</td>
<td>List of invoices for any fees that reduce the refund, such as return fees.</td>
<td>Optional</td>
<td>56.0</td>
</tr>
<tr>
<td>isAllowPartial</td>
<td>Boolean</td>
<td>This value controls the behavior when the amounts included in the sequences list don't cover the entire refund amount. If this value is false, then the default refund logic is applied to ensure the remaining refund amount. If this value is true, then the unpinned balance remains on the credit memo. If you don't specify a sequences</td>
<td>Optional</td>
<td>56.0</td>
</tr>
</tbody>
</table>
Available Version | Required or Optional | Description
--- | --- | ---
56.0 | Optional | Ordered list of refund amounts and OrderPaymentSummaries to apply them to. The process traverses this list in order and stops when it’s refunded the full amount.

**sequences**

- Type: List<ConnectApi.Sequence.OrderPaymentSummary.InputRepresentation>
- Description: List, this value is ignored and the default refund logic is applied. The default value is false.

SEE ALSO:

`ensureRefundsAsync(orderSummaryId, ensureRefundsInput)`

**ConnectApi.EntityLinkSegmentInput**

An entity link segment.

Subclass of ConnectApi.MessageSegmentInput.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>entityId</td>
<td>String</td>
<td>ID of the entity to link to. Only users with access to the entity see it. It’s hidden for users without access.</td>
<td>Required</td>
<td>43.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ExtendedFieldInput**

Contains information about the extended field.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>The name of the field, including custom field.</td>
<td>Optional</td>
<td>53.0</td>
</tr>
<tr>
<td>value</td>
<td>String</td>
<td>The value of the field.</td>
<td>Optional</td>
<td>53.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ExtensionInput**

An extension.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>alternative</td>
<td>ConnectApi.AlternativeInput</td>
<td>Alternative representation of the extension.</td>
<td>Required</td>
<td>40.0</td>
</tr>
</tbody>
</table>
### ConnectApi.ExtensionsCapabilityInput

Create or update extensions associated with a feed element.

This class is a subclass of `ConnectApi.FeedElementCapabilityInput`.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>extensionId</code></td>
<td><code>String</code></td>
<td>ID of the extension.</td>
<td>Required</td>
<td>40.0</td>
</tr>
<tr>
<td><code>payload</code></td>
<td><code>String</code></td>
<td>Payload associated with the extension.</td>
<td>Required</td>
<td>40.0</td>
</tr>
<tr>
<td><code>payloadVersion</code></td>
<td><code>String</code></td>
<td>Payload version that identifies the structure of the payload associated with the extension.</td>
<td>Optional</td>
<td>40.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- `ConnectApi.ExtensionsCapabilityInput`

### ConnectApi.ExternalCredentialInput

Input used to create or update an external credential.

**Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>authentication Protocol</td>
<td>ConnectApi. Credential Authentication Protocol</td>
<td>Authentication protocol of the external credential. Values are:</td>
<td>Required</td>
<td>58.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- AwsSv4</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Custom</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Jwt</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- OAuth</td>
<td></td>
<td></td>
</tr>
<tr>
<td>authentication ProtocolVariant</td>
<td>ConnectApi. Credential Authentication ProtocolVariant</td>
<td>Authentication protocol variant of the external credential. Values are:</td>
<td>Optional</td>
<td>58.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- ClientCredentialsClientSecret—OAuth 2.0 Client Credentials client secret.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- ClientCredentialsJwt Assertion—OAuth 2.0 Client Credentials JSON Web Token assertion.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- JwtBearer—OAuth 2.0 JSON Web Token bearer flow.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- NoAuthentication—No authentication.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>If specified, the authentication protocol variant must match the actual protocol variant of the external credential.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>customHeaders</td>
<td>List&lt;&lt;ConnectApi. CredentialCustomHeaderInput&gt;&gt;</td>
<td>List of credential custom headers.</td>
<td>Optional</td>
<td>58.0</td>
</tr>
<tr>
<td>developerName</td>
<td>String</td>
<td>Fully qualified developer name of the external credential.</td>
<td>Required</td>
<td>58.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Required for creating an external credential</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Optional for updating an external credential</td>
<td></td>
<td></td>
</tr>
<tr>
<td>masterLabel</td>
<td>String</td>
<td>External credential label.</td>
<td>Required</td>
<td>58.0</td>
</tr>
<tr>
<td>parameters</td>
<td>List&lt;&lt;ConnectApi. ExternalCredentialParameterInput&gt;&gt;</td>
<td>List of external credential parameters.</td>
<td>Optional depending on authenticationProtocol</td>
<td>58.0</td>
</tr>
<tr>
<td>Property</td>
<td>Type</td>
<td>Description</td>
<td>Required or Optional</td>
<td>Available Version</td>
</tr>
<tr>
<td>----------------</td>
<td>---------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>----------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>principals</td>
<td>List&lt;ConnectApi.ExternalCredentialPrincipalInput&gt;</td>
<td>List of principals the credential has.</td>
<td>Optional</td>
<td>58.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ExternalCredentialPrincipalInput**

External credential principal input.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>Parameter ID.</td>
<td>Optional</td>
<td>58.0</td>
</tr>
<tr>
<td>parameterName</td>
<td>String</td>
<td>Parameter name of the external credential.</td>
<td>Required</td>
<td>58.0</td>
</tr>
<tr>
<td>parameterType</td>
<td>ConnectApi.ExternalCredentialParameterType</td>
<td>Parameter type of the external credential. Values are:</td>
<td>Required</td>
<td>58.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AuthParameter</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AuthProvider</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AuthProviderUrl</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AuthProviderUrlQueryParameter</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• JwtBodyClaim</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• JwtHeaderClaim</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• SigningCertificate</td>
<td></td>
<td></td>
</tr>
<tr>
<td>parameterValue</td>
<td>String</td>
<td>Parameter value of the external credential.</td>
<td>Required</td>
<td>58.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.ExternalCredentialInput
- ConnectApi.ExternalCredentialPrincipalInput

**ConnectApi.ExternalCredentialPrincipalInput**

External credential principal input.
### ConnectApi.ExternalCredentialInput

A container for all capabilities that can be included when creating a feed element.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the principal external credential parameter.</td>
<td>Optional</td>
<td>58.0</td>
</tr>
<tr>
<td>parameters</td>
<td>List&lt;ConnectApi.ExternalCredentialParameterInput&gt;</td>
<td>List of external credential parameters.</td>
<td>Optional</td>
<td>58.0</td>
</tr>
<tr>
<td>principalName</td>
<td>String</td>
<td>Principal name.</td>
<td>Required</td>
<td>58.0</td>
</tr>
<tr>
<td>principalType</td>
<td>ConnectApi.CredentialPrincipalType</td>
<td>Type of credential principal. Values are:</td>
<td>Required</td>
<td>58.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsStsPrincipal</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NamedPrincipal</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• PerUserPrincipal</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sequenceNumber</td>
<td>Integer</td>
<td>Sequence number.</td>
<td>Required</td>
<td>58.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.ExternalCredentialInput
- ConnectApi.NamedCredentialInput

### ConnectApi.FeedElementCapabilitiesInput

A container for all capabilities that can be included when creating a feed element.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>associated Actions</td>
<td>ConnectApi.AssociatedActionsCapabilityInput</td>
<td>Describes actions added to the feed element.</td>
<td>Optional</td>
<td>33.0</td>
</tr>
<tr>
<td>bookmarks</td>
<td>ConnectApi.BookmarksCapabilityInput</td>
<td>Describes bookmarks added to the feed element.</td>
<td>Optional</td>
<td>32.0</td>
</tr>
<tr>
<td>canvas</td>
<td>ConnectApi.CanvasCapabilityInput</td>
<td>Describes a canvas app added to the feed element.</td>
<td>Optional</td>
<td>32.0</td>
</tr>
<tr>
<td>content</td>
<td>ConnectApi.ContentCapabilityInput</td>
<td>Describes content added to the feed element.</td>
<td>Optional</td>
<td>32.0–35.0</td>
</tr>
</tbody>
</table>

**Important:** This class isn’t available for feed posts in version 36.0 and later. In version 36.0 and later, use ConnectApi.FilesCapabilityInput.
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>directMessage</td>
<td>ConnectApi. DirectMessage CapabilityInput</td>
<td>Describes the direct message.</td>
<td>Optional</td>
<td>39.0</td>
</tr>
<tr>
<td>extensions</td>
<td>ConnectApi. ExtensionsCapabilityInput</td>
<td>Describes the extensions associated with the feed element.</td>
<td>Optional</td>
<td>40.0</td>
</tr>
<tr>
<td>feedEntityShare</td>
<td>ConnectApi. FeedEntityShare CapabilityInput</td>
<td>Describes the feed entity shared with the feed element.</td>
<td>Optional</td>
<td>39.0</td>
</tr>
<tr>
<td>files</td>
<td>ConnectApi. FilesCapabilityInput</td>
<td>Describes files attached to the feed element.</td>
<td>Optional</td>
<td>36.0</td>
</tr>
<tr>
<td>link</td>
<td>ConnectApi. LinkCapabilityInput</td>
<td>Describes a link added to the feed element.</td>
<td>Optional</td>
<td>32.0</td>
</tr>
<tr>
<td>poll</td>
<td>ConnectApi. PollCapabilityInput</td>
<td>Describes a poll added to the feed element.</td>
<td>Optional</td>
<td>32.0</td>
</tr>
<tr>
<td>questionAndAnswers</td>
<td>ConnectApi. QuestionAndAnswers CapabilityInput</td>
<td>Describes a question and answer capability added to the feed element.</td>
<td>Optional</td>
<td>32.0</td>
</tr>
<tr>
<td>status</td>
<td>ConnectApi. StatusCapabilityInput</td>
<td>Describes the status of the feed element.</td>
<td>Optional</td>
<td>44.0</td>
</tr>
<tr>
<td>topics</td>
<td>ConnectApi. TopicsCapabilityInput</td>
<td>Describes topics assigned to the feed element.</td>
<td>Optional</td>
<td>38.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.FeedElementInput

ConnectApi.FeedElementCapabilityInput
A feed element capability.

In API version 30.0 and earlier, most feed items can have comments, likes, topics, and so on. In version 31.0 and later, every feed item (and feed element) can have a unique set of capabilities. If a capability property exists on a feed element, that capability is available, even if the capability property doesn’t have a value. For example, if the ChatterLikes capability property exists on a feed element (with or without a value), the context user can like that feed element. If the capability property doesn’t exist, it isn’t possible to like that feed element. A capability can also contain associated data. For example, the Moderation capability contains data about moderation flags.
This class is abstract and has no public constructor. You can make an instance only of a subclass.

This class is a superclass of:

- ConnectApi.AssociatedActionsCapabilityInput
- ConnectApi.BookmarksCapabilityInput
- ConnectApi.CanvasCapabilityInput
- ConnectApi.ContentCapabilityInput
- ConnectApi.DirectMessageCapabilityInput
- ConnectApi.ExtensionsCapabilityInput
- ConnectApi.FeedEntityShareCapabilityInput
- ConnectApi.FilesCapabilityInput
- ConnectApi.LinkCapabilityInput
- ConnectApi.MuteCapabilityInput
- ConnectApi.PollCapabilityInput
- ConnectApi.QuestionAndAnswersCapabilityInput
- ConnectApi.ReadByCapabilityInput
- ConnectApi.RecordCapabilityInput
- ConnectApi.StatusCapabilityInput
- ConnectApi.TopicsCapabilityInput

ConnectApi.FeedElementInput

Feed elements are the top-level items that a feed contains. Feeds are feed element containers.

This class is abstract and has no public constructor. You can make an instance only of a subclass.

Superclass of ConnectApi.FeedItemInput.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>capabilities</td>
<td>ConnectApi.FeedElementCapabilitiesInput</td>
<td>The capabilities that define auxiliary information on this feed element.</td>
<td>Optional</td>
<td>31.0</td>
</tr>
</tbody>
</table>
| feedElementType   | ConnectApi.FeedElementType                | The type of feed element this input represents.                             | Required when creating a feed element
                                                                Optional when updating a feed element | 31.0              |
### ConnectApi.FeedEntityShareCapabilityInput

Share a feed entity with a feed post or comment.

This class is a subclass of `ConnectApi.FeedElementCapabilityInput`.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>feedEntityId</td>
<td>String</td>
<td>ID of the feed entity to share with the feed post or comment.</td>
<td>Required</td>
<td>39.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

- `ConnectApi.FeedElementCapabilitiesInput`

### ConnectApi.FeedItemInput

Used to create rich feed items, for example, feed items that include @mentions or files.

Subclass of `ConnectApi.FeedElementInput` as of version 31.0.
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>attachment</td>
<td>ConnectApi.FeedItemAttachmentInput</td>
<td>Specifies the attachment for the feed item. The feed item type is inferred based on the provided attachment. Important: As of API version 32.0, use the inherited capabilities property.</td>
<td>Optional</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>body</td>
<td>ConnectApi.MessageBodyInput</td>
<td>Message body. The body can contain up to 10,000 characters and 25 mentions. Because the character limit can change, clients should make a describeSObjects() call on the FeedItem or FeedComment object and look at the length of the Body or CommentBody field to determine the maximum number of allowed characters. If you specify originalFeedElementId to share a feed item, use the body property to add the first comment to the feed item. To edit this property in a feed item, use updateFeedElement(communityId, feedElementId, feedElement). Editing feed posts is supported in version 34.0 and later.</td>
<td>Required unless the feed item has a link capability or a content capability.</td>
<td>28.0</td>
</tr>
<tr>
<td>isBookmarkedByCurrentUser</td>
<td>Boolean</td>
<td>Specifies if the new feed item should be bookmarked for the user (true) or not (false). Important: As of API version 32.0, use the capabilities.bookmarks.isBookmarkedByCurrentUser property.</td>
<td>Optional</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>originalFeedElementId</td>
<td>String</td>
<td>To share a feed element, specify its 18-character ID. Important: As of API version 39.0, use the capabilities.feedEntityShare.feedEntityId property.</td>
<td>Optional</td>
<td>31.0–38.0</td>
</tr>
<tr>
<td>originalFeedItemId</td>
<td>String</td>
<td>To share a feed item, specify its 18-character ID. Important: In API version 32.0–38.0, use the originalFeedElementId property. In API version 39.0 and later, use the capabilities.feedEntityShare.feedEntityId property.</td>
<td>Optional</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>visibility</td>
<td>ConnectApi.FeedItemVisibilityTypeEnum</td>
<td>Type of users who can see a feed item. • AllUsers—Visibility is not limited to internal users. • InternalUsers—Visibility is limited to internal users.</td>
<td>Optional</td>
<td>28.0</td>
</tr>
</tbody>
</table>
Default values:

- For external users, the default value is AllUsers. External users must use this value to see their posts.
- For internal users, the default value is InternalUsers. Internal users can accept this value or use the value AllUsers to allow external users to see their posts.

If the parent of the feed item is a user, group, or direct message, the visibility of the feed item must be AllUsers.

---

**ConnectApi.FileInput**

Attach a file that has already been uploaded or remove a file from a feed element.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>ID of a file that has already been uploaded.</td>
<td>Required</td>
<td>36.0</td>
</tr>
<tr>
<td>operationType</td>
<td>ConnectApi.OperationType</td>
<td>Operation to carry out on the file. Values are: Add—Adds the file to the feed element. Remove—Removes the file from the feed element. Remove operations are processed before Add operations. Adding content that is already added and removing content that is already removed result in no operation.</td>
<td>Optional</td>
<td>36.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

ConnectApi.FilesCapabilityInput

---

**ConnectApi.FindRoutesWithFewestSplitsInputRepresentation**

Data used to calculate order fulfillment routes involving the fewest number of shipment splits.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>locationAvailableInventory</td>
<td>List&lt;ConnectApi.LocationAvailabilityInputRepresentation&gt;</td>
<td>Each list element represents the available quantity of a product at an inventory location.</td>
<td>At least one element is required</td>
<td>51.0</td>
</tr>
</tbody>
</table>
### maximumNumberOfSplits

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>maximumNumberOfSplits</td>
<td>Integer</td>
<td>The maximum allowable number of shipment splits. Routing options that involve more than this number of splits are not returned.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
</tbody>
</table>

*Note:* Each split represents an additional shipment. Specifying a maximum of 0 returns only locations that can fulfill the entire order in a single shipment. A maximum of 1 returns combinations of locations that can fulfill the order in one or two shipments, and so on.

### orderedQuantities

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>orderedQuantities</td>
<td>List&lt;ConnectApi.QuantityWithSkuInputRepresentation&gt;</td>
<td>Each list element represents a quantity of a product to be routed for fulfillment.</td>
<td>At least one element is required</td>
<td>51.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

*findRoutesWithFewestSplits(findRoutesWithFewestSplitsInputRepresentation)*

### ConnectApi.FindRoutesWithFewestSplitsGroupUsingOCInputRepresentation

Data used to calculate inventory availability and fulfillment routes for one order involving the fewest number of shipment splits.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>excludeLocations</td>
<td>List&lt;String&gt;</td>
<td>List of locations to exclude from the routing calculations.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>maximumNumberOfSplits</td>
<td>Integer</td>
<td>Maximum allowable number of shipment splits. Routing options that involve more than this number of splits are not returned.</td>
<td>Optional</td>
<td>54.0</td>
</tr>
</tbody>
</table>

*Note:* Each split represents an additional shipment. Specifying a maximum of 0 returns only locations that can fulfill the entire order in a single shipment. A maximum of 1 returns combinations of locations that can fulfill the order in one or two shipments, and so on.
### orderedItems

- **Type**: List<ConnectApi.FindRoutesWithFewestSplitsUsingOCIItemInputRepresentation>
- **Description**: Each list element represents a quantity of a product to be routed for fulfillment and the assigned location group or location.
- **Required or Optional**: At least one element is required

**Available Version**: 54.0

**SEE ALSO**: findRoutesWithFewestSplitsUsingOCI(findRoutesWithFewestSplitsUsingOCIInput)

### ConnectApi.FindRoutesWithFewestSplitsUsingOCIInputRepresentation

Data used to calculate order fulfillment routes involving the fewest number of shipment splits, taking into account inventory availability.

- **Property**: findRoutesWithFewestSplitsUsingOCIInputs
- **Type**: List<ConnectApi.FindRoutesWithFewestSplitsGroupUsingOCIInputRepresentation>
- **Description**: Each list element represents a routing request for one order.
- **Required or Optional**: At least one element is required

**Available Version**: 54.0

**SEE ALSO**: findRoutesWithFewestSplitsUsingOCI(findRoutesWithFewestSplitsUsingOCIInput)

### ConnectApi.FindRoutesWithFewestSplitsUsingOCIItemInputRepresentation

A quantity of a product and a location group or location assigned to fulfill it.

- **Property**: locationGroupIdentifier
  - **Type**: String
  - **Description**: The External Reference of the location group or location assigned to the order item. If you specify a location group, inventory is considered for all locations belonging to that group.
  - **Required or Optional**: Required

- **Property**: quantity
  - **Type**: Double
  - **Description**: Quantity of the product.
  - **Required or Optional**: Required

**Available Version**: 54.0
### stockKeepingUnit

**Type:** String  
SKU of the product.  
**Required or Optional:** Required  
**Available Version:** 54.0

SEE ALSO:
- `findRoutesWithFewestSplitsUsingOCI(findRoutesWithFewestSplitsUsingOCIInput)`  
- `ConnectApi.FindRoutesWithFewestSplitsUsingOCIInputRepresentation`  
- `ConnectApi.FindRoutesWithFewestSplitsGroupUsingOCIInputRepresentation`

---

**ConnectApi.FilesCapabilityInput**

Attach up to 10 files that have already been uploaded or remove one or more files from a feed element.  
This class is a subclass of `ConnectApi.FeedElementCapabilityInput`.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>items</td>
<td>List&lt;ConnectApi.FileIdInput&gt;</td>
<td>List of file IDs and operations to be carried out on those files.</td>
<td>Required</td>
<td>36.0</td>
</tr>
</tbody>
</table>

SEE ALSO:  
- `ConnectApi.FeedElementCapabilitiesInput`

---

**ConnectApi.FormFieldInput**

Marketing integration form field.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the marketing integration form field.</td>
<td>Required</td>
<td>53.0</td>
</tr>
</tbody>
</table>
| type | ConnectApi.FormFieldType | Type of marketing integration form field. Values are:  
  - Boolean  
  - Date  
  - EmailAddress  
  - Number  
  - Text | Required | 53.0 |

SEE ALSO:  
- `ConnectApi.FormInput`
ConnectApi.FormInput
Marketing integration form.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>formFieldsList</td>
<td>List&lt;ConnectApi. FormFieldInput&gt;</td>
<td>Fields for the marketing integration form.</td>
<td>Required</td>
<td>53.0</td>
</tr>
<tr>
<td>formName</td>
<td>String</td>
<td>Name of the marketing integration form.</td>
<td>Required</td>
<td>53.0</td>
</tr>
<tr>
<td>member Identification</td>
<td>Code</td>
<td>Marketing Cloud's member identification code (MID) associated with the marketing integration form.</td>
<td>Required</td>
<td>53.0</td>
</tr>
</tbody>
</table>

ConnectApi.FormSubmissionFieldInput
Marketing integration form field submission.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the marketing integration form field.</td>
<td>Required</td>
<td>53.0</td>
</tr>
<tr>
<td>value</td>
<td>String</td>
<td>Value of the marketing integration form field.</td>
<td>Required</td>
<td>53.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.FormSubmissionInput

ConnectApi.FormSubmissionInput
Marketing integration form submission.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>formFieldsList</td>
<td>List&lt;ConnectApi. FormFieldSubmissionInput&gt;</td>
<td>Fields for the marketing integration form.</td>
<td>Required</td>
<td>53.0</td>
</tr>
</tbody>
</table>

ConnectApi.FulfillmentGroupInputRepresentation
A list of OrderItemSummaries to be fulfilled together, and the fulfillment location to handle them. The fulfillment type is one of the values defined for the Type field on the FulfillmentOrder object, such as "Warehouse" or "Retail Store." The specified type is assigned to the FulfillmentOrder for this fulfillment group.
## ConnectApi.FulfillmentOrderInputRepresentation

An OrderDeliveryGroupSummary that defines a delivery method and recipient, and a list of fulfillment groups to assign to FulfillmentOrders. Each fulfillment group is a set of OrderItemSummaries that match the OrderDeliveryGroupSummary and share the same fulfillment location. The method creates a FulfillmentOrder for each fulfillment group and a FulfillmentOrderLineItem for each OrderItemSummary.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>fulfilledFromLocationId</td>
<td>String</td>
<td>ID of the fulfillment location.</td>
<td>Required</td>
<td>48.0</td>
</tr>
<tr>
<td>fulfillmentType</td>
<td>String</td>
<td>Fulfillment type. One of the Type field values defined for FulfillmentOrders.</td>
<td>Required</td>
<td>48.0</td>
</tr>
<tr>
<td>orderItemSummaries</td>
<td>List&lt;ConnectApi.OrderItemSummaryInputRepresentation&gt;</td>
<td>List of OrderItemSummaries.</td>
<td>Required</td>
<td>48.0</td>
</tr>
<tr>
<td>referenceId</td>
<td>String</td>
<td>Reference to this input for use in troubleshooting failures. This value is only used by the APIs for creating fulfillment orders for multiple order delivery group summaries.</td>
<td>Optional</td>
<td>50.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.FulfillmentOrderInputRepresentation
- createFulfillmentOrders(fulfillmentOrderInput)

## ConnectApi.FulfillmentOrderInvoiceInputRepresentation

Instantiate and include this object with no properties when creating an invoice.
This input class has no properties.

SEE ALSO:
createInvoice(fulfillmentOrderId, invoiceInput)

**ConnectApi.FulfillmentOrderLineItemInputRepresentation**

A FulfillmentOrderLineItem and quantity to cancel. You can cancel less than the full quantity, in which case you reallocate the canceled quantity to a different FulfillmentOrder.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>fulfillmentOrderLineItemId</td>
<td>String</td>
<td>ID of the FulfillmentOrderLineItem.</td>
<td>Required</td>
<td>48.0</td>
</tr>
<tr>
<td>quantity</td>
<td>Double</td>
<td>Quantity to cancel.</td>
<td>Required</td>
<td>48.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.FulfillmentOrderLineItemsToCancelInputRepresentation
cancelFulfillmentOrderLineItems(fulfillmentOrderId, cancelFulfillmentOrderLineItemsInput)

**ConnectApi.FulfillmentOrderLineItemsToCancelInputRepresentation**

A list of FulfillmentOrderLineItems and quantities to cancel.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>fulfillmentOrderLineItemsToCancel</td>
<td>List&lt;ConnectApi.FulfillmentOrderLineItemInput&gt;</td>
<td>List of FulfillmentOrderLineItems and quantities.</td>
<td>Required</td>
<td>48.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
cancelFulfillmentOrderLineItems(fulfillmentOrderId, cancelFulfillmentOrderLineItemsInput)

**ConnectApi.BaseRequest**

Base parameters for making a request to the payment gateway.

This class is abstract.

Subclass of ConnectApi.AuditParamsRequest.

Superclass of:
- ConnectApi.AuthorizationRequest
- ConnectApi.AuthorizationReversalRequest
- `ConnectApi.CaptureRequest`
- `ConnectApi.PaymentMethodTokenizationRequest`
- `ConnectApi.PostAuthRequest`
- `ConnectApi.RefundRequest`
- `ConnectApi.SaleRequest`

### `ConnectApi.GetFOCapacityValuesRequestInputRepresentation`

Locations to get fulfillment order capacity information for.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>locationIds</td>
<td><code>List&lt;String&gt;</code></td>
<td>List of IDs of the locations to get fulfillment order capacity information for.</td>
<td>Required</td>
<td>55.0</td>
</tr>
</tbody>
</table>

### `ConnectApi.GroupInformationInput`

Chatter group information input.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>text</td>
<td><code>String</code></td>
<td>The text in the “Information” section of a group.</td>
<td>28.0</td>
</tr>
<tr>
<td>title</td>
<td><code>String</code></td>
<td>The title of the “Information” section of a group.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- `ConnectApi.ChatterGroupInput`

### `ConnectApi.HashtagSegmentInput`

Include a hashtag in a feed item or comment.

Subclass of `ConnectApi.MessageSegmentInput`. 
### ConnectApi.HoldFOCapacityInputRepresentation

Request to hold fulfillment order capacity at one or more locations. Can correspond to one action call.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>holdFOCapacity Requests</td>
<td>List&lt;ConnectApi.HoldFOCapacityRequestInputRepresentation&gt;</td>
<td>List of requests to hold fulfillment order capacity at one or more locations.</td>
<td>Required</td>
<td>55.0</td>
</tr>
</tbody>
</table>

### ConnectApi.HoldFOCapacityRequestInputRepresentation

Request to hold fulfillment order capacity at one or more locations.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>allOrNothing</td>
<td>Boolean</td>
<td>Controls whether a single failed request cancels all other requests in the list (true) or whether some requests can succeed if others fail (false). The default value is false.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>capacityRequests</td>
<td>List&lt;ConnectApi.CapacityRequestInputRepresentation&gt;</td>
<td>List of requests to hold fulfillment order capacity. Each request is for one fulfillment order at one location.</td>
<td>Required</td>
<td>55.0</td>
</tr>
</tbody>
</table>

### ConnectApi.InlineImageSegmentInput

An inline image segment.
Subclass of ConnectApi.MessageSegmentInput.
### Available Version

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>altText</td>
<td>String</td>
<td>Alt text for the inline image.</td>
<td>Optional</td>
<td>35.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If not specified, the title of the inline image file is used as the alt text.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>fileId</td>
<td>String</td>
<td>ID of the inline image file.</td>
<td>Required</td>
<td>35.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- [Post a Rich-Text Feed Element with Inline Image](#)
- [ConnectApi.MessageBodyInput](#)

### ConnectApi.InnerEnsureFundsAsyncInputRepresentation

ID of an Invoice and ID of the associated OrderSummary.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>invoiceId</td>
<td>String</td>
<td>ID of the Invoice to ensure funds for.</td>
<td>Required</td>
<td>56.0</td>
</tr>
<tr>
<td>orderSummaryId</td>
<td>String</td>
<td>ID of the OrderSummary associated with the Invoice.</td>
<td>Required</td>
<td>56.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- [multipleEnsureFundsAsync(multipleEnsureFundsInput)](#)
- [ConnectApi.MultipleEnsureFundsAsyncInputRepresentation](#)

### ConnectApi.InviteInput

An invitation.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>invitees</td>
<td>List&lt;String&gt;</td>
<td>List of email addresses to send the invitation to.</td>
<td>Required</td>
<td>39.0</td>
</tr>
<tr>
<td>message</td>
<td>String</td>
<td>Message to include in the invitation.</td>
<td>Optional</td>
<td>39.0</td>
</tr>
</tbody>
</table>

### ConnectApi.InvoiceToPayInputRepresentation

Invoice for a fee.
### InvoiceId

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>invoiceId</td>
<td>String</td>
<td>ID of the invoice for a fee.</td>
<td>Required</td>
<td>56.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- `ensureRefundsAsync(orderSummaryId, ensureRefundsInput)`
- `ConnectApi.EnsureRefundsAsyncInputRepresentation`

### ConnectApi.LeadInput

Contains information about a lead or guest user.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>company</td>
<td>String</td>
<td>The company of the lead.</td>
<td>Optional</td>
<td>53.0</td>
</tr>
<tr>
<td>email</td>
<td>String</td>
<td>The email address of the lead.</td>
<td>Optional</td>
<td>53.0</td>
</tr>
<tr>
<td>extendedFields</td>
<td>List&lt;ConnectApi.ExtendedFieldInput&gt;</td>
<td>Use to add values to any of the fields, including custom fields.</td>
<td>Optional</td>
<td>53.0</td>
</tr>
<tr>
<td>firstName</td>
<td>String</td>
<td>The first name of the lead.</td>
<td>Optional</td>
<td>53.0</td>
</tr>
<tr>
<td>lastName</td>
<td>String</td>
<td>The last name of the lead.</td>
<td>Optional</td>
<td>53.0</td>
</tr>
<tr>
<td>phone</td>
<td>String</td>
<td>The phone number of the lead.</td>
<td>Optional</td>
<td>53.0</td>
</tr>
</tbody>
</table>

### ConnectApi.LinkCapabilityInput

Create or update a link on a feed element.

This class is a subclass of `ConnectApi.FeedElementCapabilityInput`.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>String</td>
<td>Link URL. The URL can be to an external site.</td>
<td>Required</td>
<td>32.0</td>
</tr>
<tr>
<td>urlName</td>
<td>String</td>
<td>Description of the link.</td>
<td>Optional</td>
<td>32.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- `ConnectApi.FeedElementCapabilitiesInput`

### ConnectApi.LinkSegmentInput

Include a link segment in a feed item or comment.

Subclass of `ConnectApi.MessageSegmentInput`.
SEE ALSO:
ConnectApi.MessageBodyInput

ConnectApi.LocationInputRepresentation
Inventory location data used to calculate shipping distance.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>countryCode</td>
<td>String</td>
<td>The country code of the location.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>locationIdentifier</td>
<td>String</td>
<td>The identifier of the location.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>postalCode</td>
<td>String</td>
<td>The postal code of the location.</td>
<td>Required</td>
<td>51.0</td>
</tr>
</tbody>
</table>

ConnectApi.LocationAvailabilityInputRepresentation
The available quantity of a product at an inventory location.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>externalReferenceId</td>
<td>String</td>
<td>The external reference ID of the inventory location.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
<tr>
<td>quantity</td>
<td>Double</td>
<td>The available quantity of the product.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>stockKeepingUnit</td>
<td>String</td>
<td>The Stock Keeping Unit of the product.</td>
<td>Required</td>
<td>51.0</td>
</tr>
</tbody>
</table>

ConnectApi.ManagedTopicPositionCollectionInput
A collection of relative positions of managed topics.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>managedTopicPositions</td>
<td>List&lt;ConnectApi.ManagedTopicPositionInput&gt;</td>
<td>List of relative positions of managed topics. This list can include Featured and Navigational managed topics and doesn’t need to include all managed topics. For more information about reordering managed topics, see the example in reorderManagedTopics(communityId, managedTopicPositionCollection).</td>
<td>Required</td>
<td>32.0</td>
</tr>
</tbody>
</table>
ConnectApi.ManagedTopicPositionInput

Relative position of a managed topic.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>managedTopicId</td>
<td>String</td>
<td>ID of existing managed topic.</td>
<td>Required</td>
<td>32.0</td>
</tr>
<tr>
<td>position</td>
<td>Integer</td>
<td>Relative position of the managed topic, indicated by zero-indexed, ascending whole numbers.</td>
<td>Required</td>
<td>32.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.ManagedTopicPositionCollectionInput

ConnectApi.MarkupBeginSegmentInput

The beginning tag for rich text markup.
Subclass of ConnectApi.MessageSegmentInput.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>altText</td>
<td>String</td>
<td>Alternative text for the Hyperlink segment.</td>
<td>Optional</td>
<td>45.0</td>
</tr>
<tr>
<td>markupType</td>
<td>ConnectApi.MarkupType</td>
<td>Type of rich text markup.</td>
<td>Required</td>
<td>35.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Bold—Bold tag.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Code—Code tag.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Hyperlink—Hyperlink anchor tag.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Italic—Italic tag.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ListItem—List item tag.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• OrderedList—Ordered list tag.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Paragraph—Paragraph tag.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Strikethrough—Strikethrough tag.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Underline—Underline tag.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• UnorderedList—Unordered list tag.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Markup segments with a markupType of Code can include only text segments.
**ConnectApiMarkupEndSegmentInput**

The end tag for rich text markup.

Subclass of ConnectApiMessageSegmentInput

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>markupType</td>
<td>ConnectApiMarkupType</td>
<td>Type of rich text markup.</td>
<td>Required</td>
<td>35.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Bold—Bold tag.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Hyperlink—Hyperlink anchor tag.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Italic—Italic tag.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- ListItem—List item tag.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- OrderedList—Ordered list tag.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Paragraph—Paragraph tag.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Strikethrough—Strikethrough tag.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Underline—Underline tag.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- UnorderedList—Unordered list tag.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- Post a Rich-Text Feed Element with Inline Image
- ConnectApi.MessageBodyInput

**ConnectApiMentionSegmentInput**

Include an @mention of a user or group in a feed post or comment. When creating a feed post or comment, you can include up to 25 mentions.

Subclass of ConnectApi.MessageSegmentInput.
### Property Description

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the user or group to mention. To mention a user, use either id or username. You can’t include both. To mention a group, you must use id.</td>
<td>28.0</td>
</tr>
<tr>
<td>username</td>
<td>String</td>
<td>User name of the user to mention. To mention a user, use either id or username. You can’t include both.</td>
<td>38.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.MessageBodyInput

### ConnectApi.MessageBodyInput

Add rich messages to feed items and comments.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>messageSegments</td>
<td>List&lt;ConnectApi.MessageSegmentInput&gt;</td>
<td>List of message segments contained in the body</td>
<td>28.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.FeedItemInput
- ConnectApi.CommentInput
- ConnectApi.AnnouncementInput

### ConnectApi.MessageSegmentInput

Used to add rich message segments to feed items and comments.

This class is abstract and has no public constructor. You can make an instance only of a subclass.

Superclass for:
- ConnectApi.EntityLinkSegmentInput
- ConnectApi.HashtagSegmentInput
- ConnectApi.InlineImageSegmentInput
- ConnectApi.LinkSegmentInput
- ConnectApi.MarkupBeginSegmentInput
- ConnectApi.MarkupEndSegmentInput
- ConnectApi.MentionSegmentInput
- ConnectApi.TextSegmentInput
Use the [ConnectApiHelper repository on GitHub](https://github.com/ConnectApiHelper) to simplify many of the tasks accomplished with `ConnectApi.MessageSegmentInput`, such as posting with inline images, rich text, and mentions.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td><code>ConnectApi.MessageSegment.Type</code></td>
<td>The type of message segment. Values are:</td>
<td>Required</td>
<td>23.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- EntityLink</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- FieldChange</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- FieldChangeName</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- FieldChangeValue</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Hashtag</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- InlineImage</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Link</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- MarkupBegin</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- MarkupEnd</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Mention</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- MoreChanges</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- ResourceLink</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Text</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

SEE ALSO:
- [Edit a Comment](#)
- [Edit a Feed Element](#)
- [Edit a Question Title and Post](#)
- [Post a Rich-Text Feed Element with Inline Image](#)
- `ConnectApi.MessageBodyInput`

### `ConnectApi.MultipleEnsureFundsAsyncInputRepresentation`

List of Invoices and the associated OrderSummaries.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>asyncInputs</td>
<td><code>List&lt;ConnectApi.InnerEnsureFundsAsyncInputRepresentation&gt;</code></td>
<td>List of Invoices to ensure funds for and the associated OrderSummaries.</td>
<td>Required</td>
<td>56.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- `multipleEnsureFundsAsync(multipleEnsureFundsInput)`
**ConnectApi.MultipleFulfillmentOrderInputRepresentation**

List of inputs for creating fulfillment orders.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>fulfillmentOrders</td>
<td>List&lt;ConnectApi.FulfillmentOrder InputRepresentation&gt;</td>
<td>Each element contains the data to create one fulfillment order.</td>
<td>Required</td>
<td>50.0</td>
</tr>
</tbody>
</table>

**ConnectApi.MultipleFulfillmentOrderInvoicesInputRepresentation**

The FulfillmentOrders to create Invoices for.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>fulfillmentOrderIds</td>
<td>List&lt;String&gt;</td>
<td>List of IDs of FulfillmentOrders to create Invoices for.</td>
<td>At least one ID is required.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

**ConnectApi.MuteCapabilityInput**

Mute or unmute a feed element.

This class is a subclass of ConnectApi.FeedElementCapabilityInput.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isMutedByMe</td>
<td>Boolean</td>
<td>Indicates whether the feed element is muted for the context user. Default value is false.</td>
<td>Required</td>
<td>35.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

setIsMutedByMe(communityId, feedElementId, isMutedByMe)

**ConnectApi.NamedCredentialInput**

Input used to create or update a named credential.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>calloutOptions</td>
<td>ConnectApi.NamedCredential.CalloutOptionsInput</td>
<td>Callout options.</td>
<td>Required</td>
<td>58.0</td>
</tr>
<tr>
<td>calloutUrl</td>
<td>String</td>
<td>URL of the named credential in a callout.</td>
<td>Required</td>
<td>58.0</td>
</tr>
</tbody>
</table>
### customHeaders

**List<ConnectApi.CredentialCustomHeaderInput>**

- **Description**: Custom HTTP headers.
- **Required or Optional**: Optional
- **Available Version**: 58.0

### developerName

**String**

- **Description**: Named credential developer name.
- **Required or Optional**: Required for creating a named credential. Optional for updating a named credential
- **Available Version**: 58.0

### externalCredentials

**List<ConnectApi.ExternalCredentialInput>**

- **Description**: External credentials used by the named credential. In version 58.0 and later only one external credential is supported.
- **Required or Optional**: Required
- **Available Version**: 58.0

### masterLabel

**String**

- **Description**: Named credential label.
- **Required or Optional**: Required
- **Available Version**: 58.0

### networkConnection

**ConnectApi.NetworkConnectionInput**

- **Description**: PrivateConnect outbound network connection.
- **Required or Optional**: Optional depending on type
- **Available Version**: 58.0

### parameters

**List<ConnectApi.NamedCredentialParameterInput>**

- **Description**: Named credential parameters.
- **Required or Optional**: Optional
- **Available Version**: 58.0

### type

**ConnectApi.NamedCredentialType**

- **Description**: Type of named credential. Values are:
  - PrivateEndpoint
  - SecuredEndpoint
- **Required or Optional**: Required
- **Available Version**: 58.0

---

**ConnectApi.NamedCredentialCalloutOptionsInput**

Named credential callout options input.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>allowMergeFieldsInBody</td>
<td>Boolean</td>
<td>Specifies whether to allow merge fields in the HTTP body (true) or not (false).</td>
<td>Required</td>
<td>58.0</td>
</tr>
<tr>
<td>allowMergeFieldsInHeader</td>
<td>Boolean</td>
<td>Specifies whether to allow merge fields in the HTTP header (true) or not (false).</td>
<td>Required</td>
<td>58.0</td>
</tr>
<tr>
<td>generateAuthorizationHeader</td>
<td>Boolean</td>
<td>Specifies whether to generate an authorization header (true) or not (false).</td>
<td>Required</td>
<td>58.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- [ConnectApi.NamedCredentialInput](#)
## ConnectApi.NamedCredentialParameterInput

Named credential parameter input.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the parameter.</td>
<td>Optional</td>
<td>58.0</td>
</tr>
<tr>
<td>parameterName</td>
<td>String</td>
<td>Description of the parameter.</td>
<td>Optional</td>
<td>58.0</td>
</tr>
<tr>
<td>parameterType</td>
<td>String</td>
<td>Name of the parameter.</td>
<td>Required</td>
<td>58.0</td>
</tr>
<tr>
<td>parameterValue</td>
<td>String</td>
<td>Type of named credential parameter. Values are:</td>
<td>Required</td>
<td>58.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- AllowedManagedPackageNamespaces</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- ClientCertificate</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Value of the parameter.</td>
<td>Required</td>
<td>58.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

- ConnectApi.NamedCredentialInput

## ConnectApi.NetworkConnectionInput

Network connection input.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>developerName</td>
<td>String</td>
<td>Developer name of the network connection.</td>
<td>Required</td>
<td>58.0</td>
</tr>
<tr>
<td>namespace</td>
<td>String</td>
<td>Namespace of the network connection.</td>
<td>Optional</td>
<td>58.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

- ConnectApi.NamedCredentialInput

## ConnectApi.NBASTrategyInput

A recommendation strategy.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>contextRecordId</td>
<td>String</td>
<td>ID of the context record. For example, if the</td>
<td>Optional</td>
<td>45.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>next best action is on a case detail page, the</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>ID of the case.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Property</td>
<td>Type</td>
<td>Description</td>
<td>Required or Optional</td>
<td>Available Version</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------</td>
<td>-------------------------------------------------------------------------------------------------------</td>
<td>----------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>maxResults</td>
<td>Integer</td>
<td>Maximum number of results. Valid values are from 1 to 25. The default is 3.</td>
<td>Optional</td>
<td>45.0</td>
</tr>
<tr>
<td>strategyContext</td>
<td>Map&lt;String, String&gt;</td>
<td>Variable and value mappings for the strategy.</td>
<td>Optional</td>
<td>45.0</td>
</tr>
<tr>
<td>debugTrace</td>
<td>Boolean</td>
<td>Specifies whether to return trace and debug information in the response (true) or not (false). If unspecified, the default is false.</td>
<td>Optional</td>
<td>45.0</td>
</tr>
</tbody>
</table>

**ConnectApi.NewUserAudienceCriteriaInput**

Criteria for the new members type of custom recommendation audience.

Subclass of `ConnectApi.AudienceCriteriaInput`.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>value</td>
<td>Double</td>
<td>The maximum number of days since a user became a site member. For example, if you specify 30, anyone who became a site member in the last 30 days is included in the new members audience.</td>
<td>Required</td>
<td>36.0</td>
</tr>
</tbody>
</table>

**ConnectApi.OAuthCredentialAuthUrlInput**

OAuth authentication flow.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>external Credential</td>
<td>String</td>
<td>Fully qualified developer name of the external credential.</td>
<td>Required</td>
<td>56.0</td>
</tr>
<tr>
<td>principalName</td>
<td>String</td>
<td>Name of the external credential named principal.</td>
<td>Required if principalType is NamedPrincipal</td>
<td>56.0</td>
</tr>
</tbody>
</table>
| principalType          | ConnectApi.CredentialPrincipalType | Type of credential principal. Values are:  
  * AwsStsPrincipal  
  * NamedPrincipal  
  * PerUserPrincipal | Required | 56.0 |
### returnUrl
**Type:** String  
**Description:** Return URL to apply to the authentication URL.  
**Required or Optional:** Optional  
**Available Version:** 56.0

### ConnectApi.OCICreateReservationInputRepresentation
Data to reserve inventory at one or more Omnichannel Inventory locations or location groups.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>actionRequestId</td>
<td>String</td>
<td>A UUID that identifies the request. Use the action request IDs in response data to identify which requests succeeded or failed.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>allowPartial</td>
<td>Boolean</td>
<td>When true, if the system can’t create the entire reservation, then it attempts to create a partial reservation.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
<tr>
<td>createRecords</td>
<td>List&lt;ConnectApi.OCICreateReservationSingleInputRepresentation&gt;</td>
<td>A list of product quantities and locations or location groups. The list can include up to 100 elements.</td>
<td>At least one element is required</td>
<td>51.0</td>
</tr>
<tr>
<td>expirationSeconds</td>
<td>Integer</td>
<td>A length of time in seconds. If the reservation isn’t fulfilled within this amount of time after the reservationTime, then it expires. The maximum value is 14400.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
<tr>
<td>externalRefId</td>
<td>String</td>
<td>External reference ID.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
<tr>
<td>reservationTime</td>
<td>String</td>
<td>The time at which to record the reservation. Example: 2020-07-24T21:13:00Z</td>
<td>Optional</td>
<td>51.0</td>
</tr>
</tbody>
</table>

### ConnectApi.OCICreateReservationSingleInputRepresentation
A quantity of a product and an Omnichannel Inventory location or location group at which to reserve it.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>locationGroupIdentifier</td>
<td>String</td>
<td>Identifier of the location group at which to reserve inventory.</td>
<td>Either locationGroupIdentifier or locationIdentifier is required, but not both</td>
<td>51.0</td>
</tr>
</tbody>
</table>
### ConnectApi.OCIReservationInputRepresentation

A list of inventory reservations to fulfill.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>fulfillmentRecords</td>
<td>List</td>
<td>A list of inventory reservations. The list can include up to 100 elements.</td>
<td>Required</td>
<td>51.0</td>
</tr>
</tbody>
</table>

### ConnectApi.OCIReservationSingleInputRepresentation

An inventory reservation to fulfill.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>actionRequestId</td>
<td>String</td>
<td>A UUID that identifies the request. Use the action request IDs in response data to identify which requests succeeded or failed.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>externalRefId</td>
<td>String</td>
<td>The external reference ID of the location that’s fulfilling the reservation.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
<tr>
<td>locationIdentifier</td>
<td>String</td>
<td>The identifier of the location that’s fulfilling the reservation.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>quantity</td>
<td>Double</td>
<td>The quantity being fulfilled.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>stockKeepingUnit</td>
<td>String</td>
<td>The SKU of the product being fulfilled.</td>
<td>Required</td>
<td>51.0</td>
</tr>
</tbody>
</table>

### ConnectApi.OCIGetInventoryAvailabilityInputRepresentation

Details of a request to retrieve inventory availability.
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>locationGroupId</code></td>
<td><code>String</code></td>
<td>The External Reference of a location group to retrieve inventory availability data for. Specifying this value retrieves inventory data for all locations belonging to this group.</td>
<td>Optional; can't combine with <code>locationGroupIdentifiers</code> or <code>locationIdentifiers</code></td>
<td>51.0</td>
</tr>
<tr>
<td><code>locationGroupIdentifiers</code></td>
<td><code>List&lt;String&gt;</code></td>
<td>A list of up to 100 External References of location groups to retrieve inventory availability data for.</td>
<td>Optional; can't combine with <code>locationGroupId</code> or <code>locationIdentifiers</code></td>
<td>51.0</td>
</tr>
<tr>
<td><code>locationIdentifiers</code></td>
<td><code>List&lt;String&gt;</code></td>
<td>A list of up to 100 External References of locations to retrieve inventory availability data for.</td>
<td>Optional; can't combine with <code>locationGroupId</code> or <code>locationIdentifiers</code></td>
<td>51.0</td>
</tr>
<tr>
<td><code>stockKeepingUnit</code></td>
<td><code>String</code></td>
<td>The SKU of a product to retrieve inventory availability data for. Specifying a SKU with no locations or location groups returns availability data for that SKU at all inventory locations that aren't assigned to location groups.</td>
<td>Optional; can't combine with <code>stockKeepingUnits</code></td>
<td>51.0</td>
</tr>
<tr>
<td><code>stockKeepingUnits</code></td>
<td><code>List&lt;String&gt;</code></td>
<td>A list of up to 100 SKUs of products to retrieve inventory availability data for.</td>
<td>Optional; can't combine with <code>stockKeepingUnit</code></td>
<td>51.0</td>
</tr>
<tr>
<td><code>useCache</code></td>
<td><code>Boolean</code></td>
<td>Whether to fetch the inventory data from the cache. The default value is True.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
</tbody>
</table>

**ConnectApi.OCIReleaseReservationInputRepresentation**

Details of one or more inventory reservations to release.

| Property                     | Type                                           | Description                                                                 | Required or Optional | Available Version |
|------------------------------|                                               |                                                                            |                      |                  |
| `releaseRecords`             | `List<ConnectApi.OCIReleaseReservationSingleInputRepresentation>` | List of inventory reservations to release. The list can include up to 100 elements. | At least one element is required. | 51.0              |

**ConnectApi.OCIReleaseReservationSingleInputRepresentation**

A single inventory reservation to release.
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>actionRequestId</td>
<td>String</td>
<td>A UUID that identifies the request. Use the action request IDs in response data to identify which requests succeeded or failed.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>externalRefId</td>
<td>String</td>
<td>The external reference ID of the location or location group that has the reservation.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
<tr>
<td>locationGroupIdentifier</td>
<td>String</td>
<td>The identifier of the location group that has the reservation.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>locationIdentifier</td>
<td>String</td>
<td>The identifier of the location that has the reservation.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>quantity</td>
<td>Double</td>
<td>The quantity of reserved inventory to release.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>stockKeepingUnit</td>
<td>String</td>
<td>The SKU of the product to release.</td>
<td>Required</td>
<td>51.0</td>
</tr>
</tbody>
</table>

**ConnectApi.OCITransferReservationInputRepresentation**

A list of inventory reservation transfers and specifies whether a single failure cancels the entire list.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>allOrNothing</td>
<td>String</td>
<td>Controls whether a single failed transfer cancels all other transfers in the transferRecords list.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
<tr>
<td>TransferId</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• To allow some transfers in the transferRecords list to succeed when others fail, don't include this property.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• To cancel all the transfers in the transferRecords list when any of them fail, set this property to a UUID. The ID must be unique, but isn't otherwise used in this version.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>transferRecords</td>
<td>List&lt;ConnectApi.OCITransferReservationSingleInputRepresentation&gt;</td>
<td>A list of inventory reservation transfers. The list can include up to 100 elements.</td>
<td>At least one element is required.</td>
<td>51.0</td>
</tr>
</tbody>
</table>
### ConnectApi.OCITransferReservationSingleInputRepresentation

An inventory reservation transfer.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>actionRequestId</td>
<td>String</td>
<td>A UUID that identifies the request. Use the action request IDs in response data to identify which requests succeeded or failed.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>externalRefId</td>
<td>String</td>
<td>The external reference ID of the location receiving the transfer.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
<tr>
<td>fromLocationGroupIdentifier</td>
<td>String</td>
<td>The identifier of the location group sending the reservation.</td>
<td>The identifier for a sending location or location group, but not both, is required</td>
<td>51.0</td>
</tr>
<tr>
<td>fromLocationIdentifier</td>
<td>String</td>
<td>The identifier of the location sending the reservation.</td>
<td>The identifier for a sending location or location group, but not both, is required</td>
<td>51.0</td>
</tr>
<tr>
<td>ignoreAvailabilityCheck</td>
<td>Boolean</td>
<td>If true, force the transfer even if the receiving location doesn’t have sufficient available inventory. The default value is false.</td>
<td>Optional</td>
<td>52.0</td>
</tr>
<tr>
<td>quantity</td>
<td>Double</td>
<td>The quantity of inventory being transferred.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>stockKeepingUnit</td>
<td>String</td>
<td>The SKU of the product being transferred.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>toLocationGroupIdentifier</td>
<td>String</td>
<td>The identifier of the location group receiving the reservation.</td>
<td>The identifier for a receiving location or location group, but not both, is required</td>
<td>51.0</td>
</tr>
<tr>
<td>toLocationIdentifier</td>
<td>String</td>
<td>The identifier of the location receiving the reservation.</td>
<td>The identifier for a receiving location or location group, but not both, is required</td>
<td>51.0</td>
</tr>
</tbody>
</table>

### ConnectApi.OrderItemSummaryInputRepresentation

An OrderItemSummary and quantity.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>orderItemSummaryId</td>
<td>String</td>
<td>ID of the OrderItemSummary.</td>
<td>Required</td>
<td>48.0</td>
</tr>
</tbody>
</table>
### ConnectApi.FulfillmentGroupInputRepresentation

**Available Version**

**Required or Optional**

**Available Version**

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>quantity</td>
<td>Double</td>
<td>Quantity to include.</td>
<td>Required</td>
<td>48.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- `ConnectApi.FulfillmentGroupInputRepresentation`
- `ConnectApi.FulfillmentOrderInputRepresentation`
- `createFulfillmentOrders(fulfillmentOrderInput)`

### ConnectApi.OrderItemSummaryAdjustmentInput

Order item summary.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>orderItemSummaryId</td>
<td>String</td>
<td>ID of the order item summary.</td>
<td>Required</td>
<td>53.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- `ConnectApi.OrderItemSummaryAdjustmentCollectionInput`

### ConnectApi.OrderItemSummaryAdjustmentCollectionInput

Collection of order item summaries.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>orderItemSummaries</td>
<td>List&lt;ConnectApi.OrderItemSummaryAdjustmentInput&gt;</td>
<td>List of order item summaries.</td>
<td>Required</td>
<td>53.0</td>
</tr>
</tbody>
</table>

### ConnectApi.OrderSummaryInputRepresentation

An order from which to create an OrderSummary. Optionally, you can specify OrderSummary-specific information such as its Status and whether it is managed in Salesforce Order Management.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>businessModel</td>
<td>String</td>
<td>The order’s business model. It can have one of these values:</td>
<td>Optional</td>
<td>53.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• B2B</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• B2C</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Available Version Required or Optional Description Type Property

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>externalReference Identifier</td>
<td>String</td>
<td>Used internally to prevent duplicate records. This value is case-sensitive.</td>
<td>Optional</td>
<td>56.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Specifies an OrderNumber to assign to the order summary.</td>
<td>Optional</td>
<td>50.0</td>
</tr>
<tr>
<td>orderId</td>
<td>String</td>
<td>ID of the original order.</td>
<td>Required</td>
<td>48.0</td>
</tr>
</tbody>
</table>
| orderLifeCycleType              | String | Specifies whether the order is managed in Salesforce Order Management or by an external system. It can have one of these values:  
  - MANAGED—Managed in Salesforce Order Management.  
  - UNMANAGED—Managed by an external system.  
  If no value is specified, the default is MANAGED. | Optional              | 49.0              |
| sourceProcess                   | String | Describes the order process creating the OrderSummary. It can have one of these values:  
  - OrderOnBehalf—An Order on Behalf Of process.  
  - Standard—Any process other than Order on Behalf Of.  
  If no value is specified, the default is Standard. | Optional              | 57.0              |
| status                          | String | Specifies a status to assign to the order summary. The value must match one of the picklist values on the Status field of the OrderSummary object. | Optional              | 50.0              |

SEE ALSO:  
createOrderSummary(orderSummaryInput)

**ConnectApi.OrderSummaryAdjustmentAggregatesAsyncInput**

Order summary IDs for calculating adjustment aggregates.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>orderSummaryIds</td>
<td>List&lt;String&gt;</td>
<td>List of order summary IDs.</td>
<td>Required</td>
<td>55.0</td>
</tr>
</tbody>
</table>
ConnectApi.OrderSummaryLookupInput

Order summary lookup input.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>orderSummaryIdOrRefNumber</td>
<td>String</td>
<td>Either the order summary ID or reference number value.</td>
<td>Required</td>
<td>58.0</td>
</tr>
<tr>
<td>verification</td>
<td>ConnectApi.OrderSummaryVerificationInput</td>
<td>Verification attributes for guest shoppers.</td>
<td>Optional</td>
<td>58.0</td>
</tr>
</tbody>
</table>

ConnectApi.OrderSummaryVerificationInput

Order summary verification input.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>email</td>
<td>String</td>
<td>Guest shopper or registered buyer’s email address.</td>
<td>Optional</td>
<td>58.0</td>
</tr>
<tr>
<td>lastName</td>
<td>String</td>
<td>Guest shopper or registered buyer’s last name.</td>
<td>Optional</td>
<td>58.0</td>
</tr>
<tr>
<td>phoneNumber</td>
<td>String</td>
<td>Guest shopper or registered buyer’s phone number.</td>
<td>Optional</td>
<td>58.0</td>
</tr>
</tbody>
</table>

ConnectApi.OrderToCartInput

Input for action adding an order to a cart.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>cartStateOrId</td>
<td>String</td>
<td>Cart state (active or current) or the ID of the cart to which the products from an order are to be copied.</td>
<td>Required</td>
<td>57.0</td>
</tr>
</tbody>
</table>

ConnectApi.PaymentGroupRequest

Payment group input consumed by a payment group service.
### ConnectApi.PaymentMethodTokenizationRequest

Payment method tokenization input consumed by the payment tokenization service.
Subclass of `ConnectApi.BaseRequest`.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>address</code></td>
<td><code>ConnectApi.AddressRequest</code></td>
<td>Address of the payment method.</td>
<td>Required</td>
<td>52.0</td>
</tr>
<tr>
<td><code>cardPaymentMethod</code></td>
<td><code>ConnectApi.CardPaymentMethodRequest</code></td>
<td>Object representation of the card payment method.</td>
<td>Required</td>
<td>52.0</td>
</tr>
<tr>
<td><code>paymentGatewayId</code></td>
<td><code>String</code></td>
<td>ID of the card payment method's payment gateway.</td>
<td>Required</td>
<td>52.0</td>
</tr>
</tbody>
</table>

### ConnectApi.PhotoInput

Specify how to crop a photo that has already been uploaded.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Available version</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>cropSize</code></td>
<td><code>Integer</code></td>
<td>The length, in pixels, of any edge of the crop square.</td>
<td>29.0</td>
</tr>
<tr>
<td><code>cropX</code></td>
<td><code>Integer</code></td>
<td>The position X, in pixels, from the left edge of the image to the start of the crop square. Top left is position (0,0).</td>
<td>29.0</td>
</tr>
<tr>
<td><code>cropY</code></td>
<td><code>Integer</code></td>
<td>The position Y, in pixels, from the top edge of the image to the start of the crop square. Top left is position (0,0).</td>
<td>29.0</td>
</tr>
<tr>
<td><code>fileId</code></td>
<td><code>String</code></td>
<td>18 character ID of an existing file. The key prefix must be 069 and the file must be an image and be smaller than 2 GB.</td>
<td>25.0</td>
</tr>
</tbody>
</table>

**Note:** Images uploaded on the Group page and on the User page don’t have file IDs and therefore can’t be used.
### ConnectApi.PinCapabilityInput

Pin or unpin a feed element to a feed.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>entityId</td>
<td>String</td>
<td>ID of the entity to pin or unpin. In version 41.0 and later, entityId must be a feed item ID. In version 41.0–42.0, only one feed item can be pinned per feed. In version 43.0 and later, three feed items can be pinned per feed.</td>
<td>Required</td>
<td>41.0</td>
</tr>
<tr>
<td>isPinned</td>
<td>Boolean</td>
<td>Specifies whether to pin (true) or unpin (false) the entity.</td>
<td>Required</td>
<td>41.0</td>
</tr>
</tbody>
</table>

### ConnectApi.PollCapabilityInput

Create, update, or vote on a poll on a feed element.

This class is a subclass of ConnectApi.FeedElementCapabilityInput.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>choices</td>
<td>List&lt;String&gt;</td>
<td>The choices used to create a new poll. You must specify 2–10 poll choices for each poll.</td>
<td>Required for creating a poll</td>
<td>32.0</td>
</tr>
<tr>
<td>myChoiceId</td>
<td>String</td>
<td>ID of an existing choice on the feed poll. Used to vote on an existing poll.</td>
<td>Required for voting on a poll</td>
<td>32.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

ConnectApi.FeedElementCapabilitiesInput
**ConnectApi.PostAuthRequest**

Payment post authorization input consumed by the payment post authorization service.

Subclass of `ConnectApi.BaseRequest`.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>accountId</td>
<td>String</td>
<td>ID of the account of the customer for the authorized payment.</td>
<td>Required</td>
<td>54.0</td>
</tr>
<tr>
<td>amount</td>
<td>Double</td>
<td>Amount of the post authorization.</td>
<td>Required</td>
<td>54.0</td>
</tr>
<tr>
<td>comments</td>
<td>String</td>
<td>Comments for payment post authorization. Maximum of 1000 characters.</td>
<td>Optional</td>
<td>54.0</td>
</tr>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Three-letter ISO 4217 currency code associated with the payment group record.</td>
<td>Optional</td>
<td>54.0</td>
</tr>
<tr>
<td>effectiveDate</td>
<td>Datetime</td>
<td>Date that the payment post authorization occurs.</td>
<td>Required</td>
<td>54.0</td>
</tr>
<tr>
<td>paymentGatewayId</td>
<td>String</td>
<td>Payment gateway that evaluates the post authorization.</td>
<td>Required</td>
<td>54.0</td>
</tr>
<tr>
<td>paymentGroup</td>
<td>`ConnectApi.</td>
<td>Payment group associated with or to be created for the request. Request must</td>
<td>Optional</td>
<td>54.0</td>
</tr>
<tr>
<td></td>
<td>PaymentGroupRequest`</td>
<td>contain either a paymentGroupId or paymentGroup, but not both.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>paymentMethod</td>
<td>`ConnectApi.</td>
<td>Payment method sent for the post authorization.</td>
<td>Required</td>
<td>54.0</td>
</tr>
<tr>
<td></td>
<td>PostAuthApi</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PaymentMethodRequest`</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**ConnectApi.PostAuthApiPaymentMethodRequest**

Payment method input for post authorization.

Subclass of `ConnectApi.BaseApiPaymentMethodRequest`.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>alternativePaymentMethod</td>
<td><code>ConnectApi. AlternativePaymentMethod</code></td>
<td>Alternative payment method.</td>
<td>Required</td>
<td>54.0</td>
</tr>
<tr>
<td>cardPaymentMethod</td>
<td><code>ConnectApi. CardPaymentMethodRequest</code></td>
<td>Card payment method.</td>
<td>Required</td>
<td>54.0</td>
</tr>
</tbody>
</table>
### ConnectApi.PricingInput

Pricing for multiple products.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>pricingLineItems</td>
<td>List&lt;ConnectApi.PricingLineItemInput&gt;</td>
<td>Up to 500 line items for pricing.</td>
<td>Required</td>
<td>49.0</td>
</tr>
</tbody>
</table>

### ConnectApi.PricingLineItemInput

Pricing line item.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>productId</td>
<td>String</td>
<td>ID of the product to price.</td>
<td>Required</td>
<td>49.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

- ConnectApi.PricingInput

### ConnectApi.ProductSearchGroupingInput

Grouping information for product search results.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>groupingOption</td>
<td>ConnectApi.CommerceSearch.GroupingOption</td>
<td>Grouping option for search results. Values are:</td>
<td>Required</td>
<td>52.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- NoGrouping—Search results aren’t grouped.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- VariationParent—Search results are grouped by the variation parent.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>topProductType</td>
<td>ConnectApi.CommerceSearch.TopProductType</td>
<td>Type of the top product to return for each product group in search results.</td>
<td>Optional</td>
<td>52.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Value is:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- VariationParent</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>If NoGrouping is specified for groupingOption, topProductType is ignored.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### ConnectApi.ProductSearchInput

Product search.
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>categoryId</td>
<td>String</td>
<td>Category ID returns results for products in this category or its subcategories.</td>
<td>52.0</td>
</tr>
<tr>
<td>fields</td>
<td>List&lt;String&gt;</td>
<td>Product fields to return in search results. Search results include fields you have access to. If unspecified, returns the Name, Description, StockKeepingUnit, ProductCode, and Family fields.</td>
<td>52.0</td>
</tr>
<tr>
<td>grouping</td>
<td>ConnectApi.ProductSearchGroupingInput</td>
<td>Specifies whether to group products in search results and how to group them. If unspecified, the default is the value specified in Search &gt; Results Display Settings &gt; Results Grouping.</td>
<td>52.0</td>
</tr>
<tr>
<td>includePrices</td>
<td>Boolean</td>
<td>Specifies whether to include prices for products in search results (true) or not (false). If unspecified, defaults to false.</td>
<td>52.0</td>
</tr>
<tr>
<td>includeQuantityRule</td>
<td>Boolean</td>
<td>Specifies whether to include purchase quantity rule information for products in search results (true) or not (false). If unspecified, defaults to false.</td>
<td>52.0</td>
</tr>
<tr>
<td>page</td>
<td>Integer</td>
<td>Number of the page you want returned. Starts at 0. If you pass in null or 0, the first page is returned.</td>
<td>52.0</td>
</tr>
<tr>
<td>pageSize</td>
<td>Integer</td>
<td>Specifies the number of items per page. Valid values are from 1 through 200. If unspecified, the default is the value specified in Results per Page in Search &gt; Results Display Settings.</td>
<td>52.0</td>
</tr>
<tr>
<td>refinements</td>
<td>List&lt;ConnectApi.RefinementInput&gt;</td>
<td>List up to nine refinements (facets) for search results. Buyers or shoppers can select up to 20 values for each refinement.</td>
<td>52.0</td>
</tr>
<tr>
<td>searchTerm</td>
<td>String</td>
<td>List of up to 32 space-separated search terms.</td>
<td>52.0</td>
</tr>
<tr>
<td>sortRuleId</td>
<td>String</td>
<td>ID of the sort rule that specifies the order of products in the search results.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

**ConnectApi.PromotionCartInput**

Cart during promotion evaluation.
### ConnectApi.PromotionCartInput

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>cartItems</td>
<td>List&lt;ConnectApi.PromotionCartItemInput&gt;</td>
<td>A collection of items in the cart.</td>
<td>Required</td>
<td>57.0</td>
</tr>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Three-letter ISO 4217 currency code associated with the cart.</td>
<td>Required for multi-currency orgs</td>
<td>57.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the cart.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.PromotionEvaluateInput
  - evaluate(salesTransaction)

#### ConnectApi.PromotionCartDeliveryGroupInput

IDs of the cart delivery group and its delivery method.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>cartDeliveryGroupId</td>
<td>String</td>
<td>ID of the cart delivery group.</td>
<td>Required</td>
<td>57.0</td>
</tr>
<tr>
<td>deliveryMethodId</td>
<td>String</td>
<td>ID of the order delivery method.</td>
<td>Required</td>
<td>57.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.PromotionEvaluateInput
  - evaluate(salesTransaction)

#### ConnectApi.PromotionCartItemInput

Item in a cart during promotion evaluation.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>cartDeliveryGroupId</td>
<td>String</td>
<td>ID of the cart delivery group.</td>
<td>Required when evaluating shipping promotions</td>
<td>57.0</td>
</tr>
<tr>
<td>cartId</td>
<td>String</td>
<td>ID of the cart.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the cart item. Must be unique across all items in the cart.</td>
<td>Required</td>
<td>57.0</td>
</tr>
<tr>
<td>itemDescription</td>
<td>String</td>
<td>Description of the cart item.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
</tbody>
</table>
### Available Version

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>itemName</td>
<td>String</td>
<td>Name of the cart item.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>listPrice</td>
<td>String</td>
<td>List price of the cart item.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>product2Id</td>
<td>String</td>
<td>Product ID of the cart item.</td>
<td>Required if SKU isn't specified</td>
<td>57.0</td>
</tr>
<tr>
<td>quantity</td>
<td>String</td>
<td>Number of items in the cart.</td>
<td>Required</td>
<td>57.0</td>
</tr>
<tr>
<td>salesPrice</td>
<td>String</td>
<td>Sales price of the cart item. This is the price per quantity and the value used to compute the discount. If salesPrice and totalLineBaseAmount are specified, totalLineBaseAmount is used.</td>
<td>Required if salesPrice isn't specified</td>
<td>57.0</td>
</tr>
<tr>
<td>sku</td>
<td>String</td>
<td>Stock keeping unit (SKU) of the cart item.</td>
<td>Required if product2Id isn't specified</td>
<td>57.0</td>
</tr>
<tr>
<td>totalLineBaseAmount</td>
<td>String</td>
<td>Total amount for the cart item, equal to sales price multiplied by quantity. This value is used to compute the discount. If salesPrice and totalLineBaseAmount are specified, totalLineBaseAmount is used.</td>
<td>Required if salesPrice isn't specified</td>
<td>57.0</td>
</tr>
<tr>
<td>totalListBaseAmount</td>
<td>String</td>
<td>Total amount for the cart item based on list price and quantity.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>type</td>
<td>ConnectApi.CartItemType</td>
<td>Type of item in a cart. Values are: DeliveryCharge, Product</td>
<td>Required when evaluating shipping promotions</td>
<td>57.0</td>
</tr>
</tbody>
</table>

### SEE ALSO:
- ConnectApi.PromotionCartInput
- ConnectApi.PromotionEvaluateInput
evaluate(salesTransaction)

### ConnectApi.PromotionEvaluateInput

Find promotions that the customer is eligible for and compute their discounts.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>cart</td>
<td>ConnectApi.PromotionCartInput</td>
<td>Cart and its items.</td>
<td>Required</td>
<td>57.0</td>
</tr>
</tbody>
</table>
### ConnectApi.PromotionCartDeliveryGroups

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>cartDeliveryGroups</td>
<td>List&lt;ConnectApi.PromotionCartDeliveryGroup&gt;</td>
<td>List of cart delivery groups associated with the items in the cart. Available if shipping promotions are enabled.</td>
<td>Required when evaluating shipping promotions</td>
<td>57.0</td>
</tr>
<tr>
<td>couponCodes</td>
<td>List&lt;String&gt;</td>
<td>List of coupon codes to enable promotions. A customer can apply a maximum of two coupons per cart.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>effectiveAccountId</td>
<td>String</td>
<td>ID of the account for which the request is made.</td>
<td>Required if segments isn’t specified</td>
<td>57.0</td>
</tr>
<tr>
<td>isItemizeHeaderAdjustments</td>
<td>Boolean</td>
<td>Specifies whether order-level adjustments are itemized (true) or not (false). If unspecified, the default value is false.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>parentProducts</td>
<td>List&lt;ConnectApi.PromotionParentProductsInput&gt;</td>
<td>Map of the parent product ID to its variation product IDs.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>productCategories</td>
<td>List&lt;ConnectApi.PromotionProductCategoriesInput&gt;</td>
<td>Map of product IDs to their associated category IDs.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>segments</td>
<td>List&lt;String&gt;</td>
<td>All promotions associated with promotion segments specified in this list are active and can be evaluated against the cart. Additionally, any segments associated with a store or buyer group are also still evaluated against the cart. If this field is not present, only the promotions associated with a store or buyer group are evaluated.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
<tr>
<td>webStoreId</td>
<td>String</td>
<td>ID of the store for which the request is made. If unspecified, defined segments must be used instead.</td>
<td>Optional</td>
<td>57.0</td>
</tr>
</tbody>
</table>

See also:

evaluate(salesTransaction)

### ConnectApi.PromotionParentProductsInput

IDs of a parent product and variation product.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>childProductId</td>
<td>String</td>
<td>ID of the variation product.</td>
<td>Required</td>
<td>57.0</td>
</tr>
</tbody>
</table>
### ConnectApi.PromotionEvaluateInput

IDs of a product and associated category.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>parentProductId</td>
<td>String</td>
<td>ID of the parent product.</td>
<td>Required</td>
<td>57.0</td>
</tr>
<tr>
<td>categoryId</td>
<td>String</td>
<td>ID of the category.</td>
<td>Required</td>
<td>57.0</td>
</tr>
<tr>
<td>productId</td>
<td>String</td>
<td>ID of the product.</td>
<td>Required</td>
<td>57.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- `ConnectApi.PromotionEvaluateInput` evaluate(salesTransaction)

### ConnectApi.PromotionProductCategoriesInput

IDs of a product and associated category.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>categoryId</td>
<td>String</td>
<td>ID of the category.</td>
<td>Required</td>
<td>57.0</td>
</tr>
<tr>
<td>productId</td>
<td>String</td>
<td>ID of the product.</td>
<td>Required</td>
<td>57.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- `ConnectApi.PromotionEvaluateInput` evaluate(salesTransaction)

### ConnectApi.QuantityWithSkuInputRepresentation

A quantity of a product.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>quantity</td>
<td>Double</td>
<td>Quantity of the product.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>stockKeepingUnit</td>
<td>String</td>
<td>SKU of the product.</td>
<td>Required</td>
<td>51.0</td>
</tr>
</tbody>
</table>

### ConnectApi.QuestionAndAnswersCapabilityInput

Create or edit a question feed element or set the best answer of the existing question feed element.

This class is a subclass of `ConnectApi.FeedElementCapabilityInput`. 
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>bestAnswerId</td>
<td>String</td>
<td>A comment ID to use as a best answer for a question feed element. The best answer comment must already exist on the question feed element.</td>
<td>Required to update a feed element. Not supported when posting a feed element.</td>
<td>32.0</td>
</tr>
<tr>
<td>questionTitle</td>
<td>String</td>
<td>Title for a question feed element. To edit the title of a question, use <code>updateFeedElement(communityId, feedElementId, feedElement)</code>. Editing question titles is supported in version 34.0 and later.</td>
<td>Required to post a feed element. Not supported when updating a feed element.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- Edit a Question Title and Post
- `ConnectApi.FeedElementCapabilitiesInput`

### ConnectApi.RankAverageDistanceInputRepresentation

An order recipient's geographic location and information about sets of inventory locations that can fulfill the order.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>deliveryCountryCode</td>
<td>String</td>
<td>The country code of the order recipient.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>deliveryPostalCode</td>
<td>String</td>
<td>The postal code of the order recipient.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>distanceUnit</td>
<td>String</td>
<td>Specify <code>mi</code> or <code>km</code> to return average distances in miles or kilometers, respectively.</td>
<td>51.0</td>
<td></td>
</tr>
<tr>
<td>sortResult</td>
<td>String</td>
<td>Specify <code>ASC</code> or <code>DESC</code> to rank the results by average shipping distance in ascending or descending order, respectively.</td>
<td>51.0</td>
<td></td>
</tr>
<tr>
<td>targetLocations</td>
<td>List&lt;<code>ConnectApi.TargetLocationInputRepresentation&gt;</code></td>
<td>Each element is a set of inventory locations that can combine to fulfill the order.</td>
<td>At least one element is required</td>
<td>51.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ReadByCapabilityInput

Mark feed elements as read by the context user.

This class is a subclass of `ConnectApi.FeedElementCapabilityInput`. 
### isReadByMe
- **Type:** Boolean
- **Description:** Specifies to mark the feed element as read (true) for the context user.
- **Required or Optional:** Required
- **Available Version:** 40.0

### lastReadDateByMe
- **Type:** Datetime
- **Description:** Specifies the last date, in ISO 8601 format, when the feed element is marked as read for the context user. If you don’t specify a date or you specify a future date, the current system date is used.
- **Required or Optional:** Optional
- **Available Version:** 40.0

SEE ALSO:
- `ConnectApi.FeedElementCapabilitiesInput`

---

### ConnectApi.SequenceOrderPaymentSummaryInputRepresentation

Amount to apply to specified OrderPaymentSummary as part of a payment or refund.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>amount</td>
<td>Double</td>
<td>Amount to apply to the OrderPaymentSummary.</td>
<td>Optional</td>
<td>56.0</td>
</tr>
<tr>
<td>orderPaymentSummaryId</td>
<td>String</td>
<td>ID of the OrderPaymentSummary to apply the Amount to.</td>
<td>Required</td>
<td>56.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- `ensureRefundsAsync(orderSummaryId, ensureRefundsInput)`
- `ConnectApi.EnsureRefundsAsyncInputRepresentation`

---

### ConnectApi.RecipientEngagementContextInput

Context based on which the survey invitation is sent to a participant.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>recipientEngagementContext</td>
<td>Map&lt;String, String&gt;</td>
<td>Map each recipient with the context based on which the survey invitation is emailed.</td>
<td>Required</td>
<td>50.0</td>
</tr>
</tbody>
</table>
## ConnectApi.RecommendationAudienceInput

A custom recommendation audience.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>recipientId</td>
<td>String</td>
<td>Participant ID with whom the engagement context should be associated.</td>
<td>Required</td>
<td>50.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- `ConnectApi.SurveyInvitationEmailInput`
- `ConnectApi.RecommendationAudienceInput`

### ConnectApi.RecommendationAudienceInput

A custom recommendation audience.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>criteria</td>
<td><code>ConnectApi.AudienceCriteriaInput</code></td>
<td>The criteria for the custom recommendation audience type.</td>
<td>Optional</td>
<td>36.0</td>
</tr>
</tbody>
</table>

**Important:** This property is only available in version 35.0. In version 36.0 and later, use `ConnectApi.CustomListAudienceCriteriaInput`.

<table>
<thead>
<tr>
<th>memberOperation Type</th>
<th><code>ConnectApi.RecommendationAudienceMemberOperationType</code></th>
<th>The operation to carry out on the audience members.</th>
<th>Required</th>
<th>35.0 only</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>• Add—Adds specified members to the audience.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Remove—Removes specified members from the audience.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Important:** This property is available only in version 35.0. In version 36.0 and later, use `ConnectApi.CustomListAudienceCriteriaInput`.

| members                   | `List<String>` | A collection of user IDs. When updating an audience, you can include up to 100 members. An audience can have up to 100,000 members, and each | Required             | 35.0 only         |

**Important:** This property is available only in version 35.0. In version 36.0 and later, use `ConnectApi.CustomListAudienceCriteriaInput`.
Experience Cloud site can have up to 100 audiences.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>The unique name of the custom recommendation audience.</td>
<td>Optional to update a recommendation audience</td>
<td>35.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Required to create a recommendation audience</td>
<td></td>
</tr>
</tbody>
</table>

**SEE ALSO:**

`createRecommendationAudience(communityId, recommendationAudience)`

**ConnectApi.RecommendationDefinitionInput**

A custom recommendation definition.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>actionUrl</td>
<td>String</td>
<td>URL for acting on the custom recommendation, for example, the URL to join a group.</td>
<td>Required to create a recommendation definition</td>
<td>35.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Optional to update a recommendation definition</td>
<td></td>
</tr>
<tr>
<td>actionUrlName</td>
<td>String</td>
<td>Text label for the action URL in the user interface, for example, “Launch.”</td>
<td>Required to create a recommendation definition</td>
<td>35.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Optional to update a recommendation definition</td>
<td></td>
</tr>
<tr>
<td>explanation</td>
<td>String</td>
<td>Explanation, or body, of the custom recommendation.</td>
<td>Required to create a recommendation definition</td>
<td>35.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Optional to update a recommendation definition</td>
<td></td>
</tr>
</tbody>
</table>
### Available Version
Required or Optional
Available Version

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the custom recommendation definition. The name is displayed in Setup.</td>
<td>Required to create a recommendation definition</td>
<td>35.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Optional to update a recommendation definition</td>
<td></td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>Title of the custom recommendation definition.</td>
<td>Optional</td>
<td>35.0</td>
</tr>
</tbody>
</table>

#### SEE ALSO:
createRecommendationDefinition(communityId, recommendationDefinition)

### ConnectApi.RecommendationReactionInput
A reaction to a recommendation produced by a recommendation strategy.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>aiModel</td>
<td>String</td>
<td>Reserved for future use.</td>
<td>Optional</td>
<td>47.0</td>
</tr>
<tr>
<td>contextRecordId</td>
<td>String</td>
<td>ID of the context record. For example, if the next best action is on a case detail page, the ID of the case.</td>
<td>Optional</td>
<td>45.0</td>
</tr>
<tr>
<td>executionId</td>
<td>String</td>
<td>ID of the original recommendation strategy execution.</td>
<td>Optional</td>
<td>45.0</td>
</tr>
<tr>
<td>externalId</td>
<td>String</td>
<td>External ID of the recommendation. This ID doesn't need to be a Salesforce 18-character ID. For example, it can be a product number from an external system.</td>
<td>Optional</td>
<td>46.0</td>
</tr>
<tr>
<td>onBehalfOfId</td>
<td>String</td>
<td>ID of the user or entity for which the reaction took place.</td>
<td>Optional</td>
<td>45.0</td>
</tr>
<tr>
<td>reactionType</td>
<td>ConnectApi.RecommendationReactionType</td>
<td>Type of reaction to a recommendation. Values are:</td>
<td>Required</td>
<td>45.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Accepted</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Rejected</td>
<td></td>
<td></td>
</tr>
<tr>
<td>recommendationMode</td>
<td>String</td>
<td>Reserved for future use.</td>
<td>Optional</td>
<td>46.0</td>
</tr>
<tr>
<td>recommendationScore</td>
<td>Double</td>
<td>Reserved for future use.</td>
<td>Optional</td>
<td>46.0</td>
</tr>
</tbody>
</table>
### ConnectApi.RecordCapabilityInput

Attach an existing knowledge article to a comment.

This class is a subclass of `ConnectApi.FeedElementCapabilityInput`.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>recordId</td>
<td>String</td>
<td>ID of the existing knowledge article to attach.</td>
<td>Required</td>
<td>42.0</td>
</tr>
</tbody>
</table>

### ConnectApi.RecordsetFilterCriteriaInput

A set of recordset filter criteria applied to records, such as service appointment records.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>criteriaIds</td>
<td>List&lt;String&gt;</td>
<td>Recordset filter criteria IDs.</td>
<td>Required</td>
<td>53.0</td>
</tr>
<tr>
<td>enforceSharing</td>
<td>Boolean</td>
<td>Determines whether record sharing checks are enforced (true) or not (false) during the execution of this call.</td>
<td>Optional</td>
<td>53.0</td>
</tr>
<tr>
<td>filteredObjectName</td>
<td>String</td>
<td>Object that the filter is applied to.</td>
<td>Required</td>
<td>53.0</td>
</tr>
<tr>
<td>recordIds</td>
<td>List&lt;String&gt;</td>
<td>List of record IDs of the filtered object.</td>
<td>Required</td>
<td>53.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ReferencedRefundRequest

Referenced refund input.

Subclass of `ConnectApi.RefundRequest`.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>accountId</td>
<td>String</td>
<td>ID of the account linked to the referenced refund request.</td>
<td>Optional</td>
<td>50.0</td>
</tr>
</tbody>
</table>
### ConnectApi.RefinementInput

Attribute-based refinement input for product search.

This class is abstract and is a superclass of `ConnectApi.DistinctValueRefinementInput`.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>attributeType</td>
<td><code>ConnectApi.CommerceSearch.AttributeType</code></td>
<td>Search attribute type.</td>
<td>Required</td>
<td>52.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Custom</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- ProductAttribute</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Standard</td>
<td></td>
<td></td>
</tr>
<tr>
<td>nameOrId</td>
<td>String</td>
<td>Developer name of the attribute.</td>
<td>Required</td>
<td>52.0</td>
</tr>
<tr>
<td>type</td>
<td><code>ConnectApi.CommerceSearch.FacetType</code></td>
<td>Search facet type. Value is:</td>
<td>Required</td>
<td>52.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- DistinctValue</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### ConnectApi.RefundRequest

Refund input.

This class is abstract.

This class is a subclass of `ConnectApi.BaseRequest`.

No additional properties.

This class is a superclass of `ConnectApi.ReferencedRefundRequest`.

### ConnectApi.ReleaseHeldFOCapacityInputRepresentation

Request to release held fulfillment order capacity at one or more locations. Can correspond to one action call.
### ConnectApi.ReleaseHeldFOCapacityRequestInputRepresentation

Request to release held fulfillment order capacity at one or more locations.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>releaseHeldFOCapacityRequests</td>
<td>List&lt;ConnectApi.ReleaseHeldFOCapacityRequestInputRepresentation&gt;</td>
<td>List of requests to release held fulfillment order capacity at one or more locations.</td>
<td>Required</td>
<td>55.0</td>
</tr>
<tr>
<td>allOrNothing</td>
<td>Boolean</td>
<td>Controls whether a single failed request cancels all other requests in the list (<code>true</code>) or whether some requests can succeed if others fail (<code>false</code>). The default value is <code>false</code>.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>capacityRequests</td>
<td>List&lt;ConnectApi.CapacityRequestInputRepresentation&gt;</td>
<td>List of requests to release held fulfillment order capacity. Each request is for capacity for one fulfillment order held at one location.</td>
<td>Required</td>
<td>55.0</td>
</tr>
</tbody>
</table>

### ConnectApi.RequestHeaderInput

An HTTP request header name and value pair.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>The name of the request header.</td>
<td>Required</td>
<td>33.0</td>
</tr>
<tr>
<td>value</td>
<td>String</td>
<td>The value of the request header.</td>
<td>Required</td>
<td>33.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

Define an Action Link and Post with a Feed Element

### ConnectApi.ReturnItemsInputRepresentation

Data about products and delivery charges to return, as well as associated return fees.
### ConnectApi.ReturnOrderInputRepresentation

Data for creating a ReturnOrder and ReturnOrderLineItems.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>orderSummaryId</td>
<td>String</td>
<td>ID of the OrderSummary containing the items to be returned. The OrderSummary’s OrderLifeCycleType must be Managed.</td>
<td>Required</td>
<td>50.0</td>
</tr>
<tr>
<td>returnOrderLifeCycleType</td>
<td>String</td>
<td>The LifeCycleType of the ReturnOrder. Possible values are:</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Managed—Process the ReturnOrder using the APIs and actions. It can generate change orders and affects financial fields and rollup calculations.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Unmanaged—The ReturnOrder is for tracking purposes only. It isn’t involved in any financial calculations and doesn’t generate any change orders. The system doesn’t prevent the creation of duplicate ReturnOrderLineItems in an unmanaged ReturnOrder for the same OrderItem.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>returnOrderLineItems</td>
<td>List&lt;String&gt;</td>
<td>List of data for creating ReturnOrderLineItems.</td>
<td>At least one element is required</td>
<td>50.0</td>
</tr>
<tr>
<td>Property</td>
<td>Type</td>
<td>Description</td>
<td>Required or Optional</td>
<td>Available Version</td>
</tr>
<tr>
<td>---------------</td>
<td>------------</td>
<td>------------------------------------------------------------------------------</td>
<td>----------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>status</td>
<td>String</td>
<td>Status to assign the ReturnOrder. This value must match an entry in the ReturnOrder object’s Status picklist.</td>
<td>Required</td>
<td>51.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

createReturnOrder(returnOrderInput)

**ConnectApi.ReturnOrderItemInputRepresentation**

ID of a ReturnOrderLineItem and instructions for updating it. After the update, the ReturnOrderLineItem is read-only. Any remaining quantity to be returned is added to a new ReturnOrderLineItem.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>quantityReceived</td>
<td>Double</td>
<td>The quantity of the ReturnOrderLineItem that has been received. The value must be zero or greater. This value isn’t used by any standard features, but is provided for use in customizations.</td>
<td>Required</td>
<td>52.0</td>
</tr>
<tr>
<td>quantityRejected</td>
<td>Double</td>
<td>The quantity of the ReturnOrderLineItem that has been rejected for return. The value must be zero or greater. This value isn’t used by any standard features, but is provided for use in customizations.</td>
<td>Required</td>
<td>52.0</td>
</tr>
<tr>
<td>quantityReturned</td>
<td>Double</td>
<td>The quantity of the ReturnOrderLineItem that has been returned. The value must be greater than zero. If this value plus quantityToCancel is less than the expected return quantity, then the remaining quantity to be returned is added to a new ReturnOrderLineItem.</td>
<td>Required</td>
<td>52.0</td>
</tr>
<tr>
<td>quantityToCancel</td>
<td>Double</td>
<td>The quantity of the ReturnOrderLineItem to remove because it’s not being returned. The value must be zero or greater. If this value plus quantityReturned is less than the expected return quantity, then the remaining quantity to be returned is added to a new ReturnOrderLineItem.</td>
<td>Required</td>
<td>52.0</td>
</tr>
<tr>
<td>reasonForRejection</td>
<td>String</td>
<td>The reason why the rejected quantity, if any, was rejected. This value isn’t used by any standard features, but is provided for use in customizations.</td>
<td>Optional</td>
<td>52.0</td>
</tr>
</tbody>
</table>
### ConnectApi.ReturnOrderLineItemDeliveryChargeInputRepresentation

ID of a ReturnOrderLineItem representing a delivery charge.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>returnOrderLineItemId</td>
<td>String</td>
<td>ID of a ReturnOrderLineItem to return.</td>
<td>Required</td>
<td>52.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ReturnOrderItemFeeInputRepresentation

ID of a ReturnOrderLineItem representing a return fee, and instructions for updating it. After the update, the ReturnOrderLineItem is read-only. Any remaining quantity of the fee to be processed is added to a new ReturnOrderLineItem.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>quantityReturned</td>
<td>Double</td>
<td>Quantity of the ReturnOrderLineItem to process. When the fee is a fixed amount, the charge is determined by multiplying the total fee amount by this value divided by the expected quantity. For example, if the fee amount is $10 and the expected quantity is 2, then if the quantityReturned is 1, $5 is charged. This value normally equals the quantity returned of the ReturnOrderLineItem for the returned item that the fee applies to. The value must be greater than zero. If this value plus quantityToCancel is less than the expected quantity, then the remaining</td>
<td>Required</td>
<td>56.0</td>
</tr>
<tr>
<td>Property</td>
<td>Type</td>
<td>Description</td>
<td>Required or Optional</td>
<td>Available Version</td>
</tr>
<tr>
<td>-----------------------</td>
<td>----------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>quantityToCancel</td>
<td>Double</td>
<td>Quantity of the ReturnOrderLineItem to remove. This value normally equals the quantity canceled of the ReturnOrderLineItem for the returned item that the fee applies to. This value can also be used to cancel a portion of the fee. The value must be zero or greater. If this value plus quantityReturned is less than the expected quantity, then the remaining quantity to be returned is added to a new ReturnOrderLineItem.</td>
<td>Required</td>
<td>56.0</td>
</tr>
<tr>
<td>returnOrderLineItemId</td>
<td>String</td>
<td>ID of the ReturnOrderLineItem representing the return fee.</td>
<td>Required</td>
<td>56.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- `ConnectApi.ReturnItemsInputRepresentation`
- `returnItems(returnOrderId, returnItemsInput)`

### ConnectApi.ReturnOrderLineItemInputRepresentation

Data for creating a ReturnOrderLineItem for an order item being returned, including data to create ReturnOrderLineItems representing any return fees associated with it.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>canReduceShipping</td>
<td>Boolean</td>
<td>Whether to refund any associated shipping charge.</td>
<td>Required</td>
<td>50.0</td>
</tr>
<tr>
<td>orderItemSummaryId</td>
<td>String</td>
<td>ID of the associated OrderItemSummary. If the OrderItemSummary already has an associated ReturnOrderLineItem, then you must specify a different reasonForReturn. Duplicating the reason breaks the financial calculations.</td>
<td>Required</td>
<td>50.0</td>
</tr>
<tr>
<td>quantityExpected</td>
<td>Double</td>
<td>Quantity expected to be returned. This value also applies to any fees specified in returnOrderLineItemFees.</td>
<td>Required</td>
<td>50.0</td>
</tr>
<tr>
<td>quantityReceived</td>
<td>Double</td>
<td>Quantity already physically returned.</td>
<td>Optional</td>
<td>50.0</td>
</tr>
</tbody>
</table>
## reasonForReturn

**Type:** String  
**Description:** Reason for the return. The value must match an entry in both the OrderSummaryChange Reason field and the ReturnOrderLineItem object's ReasonForReturn picklist.  
**Required or Optional:** Required if the returnOrder LifeCycleType is MANAGED.  
**Available Version:** 50.0

## returnOrderLineItemFees

**Type:** List\<\<ConnectApi.ReturnOrderLineItemFeeInputRepresentation\>>  
**Description:** List of input data for return fees associated with the order item being returned. A ReturnOrderLineItem of Type Fee is created to represent each fee.  
**Required or Optional:** Optional  
**Available Version:** 56.0

---

### SEE ALSO:

- ConnectApi.ReturnOrderInputRepresentation
- createReturnOrder(returnOrderInput)

### ConnectApi.ReturnOrderLineItemFeeInputRepresentation

Data for creating a ReturnOrderLineItem that represents a return fee.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>amount</td>
<td>Double</td>
<td>Value used to calculate the fee amount, as described by the amountType. It must be a positive value.</td>
<td>Required</td>
<td>56.0</td>
</tr>
<tr>
<td>amountType</td>
<td>String</td>
<td>Describes how the fee amount is calculated. It can have one of these values:</td>
<td>Required</td>
<td>56.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AmountWithTax—Value of amount is the fee amount, including tax.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AmountWithoutTax—Value of amount is the fee amount, not including tax. Tax is calculated on the value and added.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Percentage—Value of amount is a percentage. To determine the fee amount, amount is divided by 100, and then multiplied by the TotalPrice and TotalTaxAmount of the associated OrderItemSummary, prorated for the quantity being returned.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• PercentageGross—Value of amount is a percentage. To determine the fee amount, amount is divided by 100, and then multiplied by the</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### ConnectApi.ReturnOrderInputRepresentation

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the fee.</td>
<td>Required</td>
<td>56.0</td>
</tr>
<tr>
<td>product2Id</td>
<td>String</td>
<td>ID of the product representing the fee.</td>
<td>Required</td>
<td>56.0</td>
</tr>
<tr>
<td>reason</td>
<td>String</td>
<td>Reason for the fee. The value must match an entry in the ReturnOrderLineItem object's ReasonForReturn picklist.</td>
<td>Required</td>
<td>56.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.ReturnOrderInputRepresentation
- createReturnOrder(returnOrderInput)
- ConnectApi.ReturnOrderLineItemInputRepresentation

### ConnectApi.SaleRequest

Payment sale input consumed by the payment sale service.

Subclass of ConnectApi.BaseRequest.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>accountId</td>
<td>String</td>
<td>Reference to account.</td>
<td>Required</td>
<td>54.0</td>
</tr>
<tr>
<td>amount</td>
<td>Double</td>
<td>The amount of the sale request.</td>
<td>Required</td>
<td>54.0</td>
</tr>
<tr>
<td>comments</td>
<td>String</td>
<td>Optional comment for the sale request.</td>
<td>Optional</td>
<td>54.0</td>
</tr>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Three-letter ISO 4217 currency code associated with the payment output.</td>
<td>Required</td>
<td>54.0</td>
</tr>
<tr>
<td>effectiveDate</td>
<td>Datetime</td>
<td>Date that the sale request takes effect.</td>
<td>Required</td>
<td>54.0</td>
</tr>
<tr>
<td>paymentGatewayId</td>
<td>String</td>
<td>The payment gateway that receives the sale request.</td>
<td>Required</td>
<td>54.0</td>
</tr>
<tr>
<td>paymentGroup</td>
<td>ConnectApi.PaymentGroupRequest</td>
<td>Payment group information for the sale request.</td>
<td>Optional</td>
<td>54.0</td>
</tr>
<tr>
<td>paymentMethod</td>
<td>ConnectApi.SaleApiPaymentMethodRequest</td>
<td>Payment method used within the sale request.</td>
<td>Required</td>
<td>54.0</td>
</tr>
</tbody>
</table>
### ConnectApi.SaleApiPaymentMethodRequest

Payment method request for sale.

Subclass of [ConnectApi.BaseApiPaymentMethodRequest](#).

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>cardPaymentMethod</td>
<td><a href="#">ConnectApi.CardPaymentMethodRequest</a></td>
<td>Payment method used in a sale request.</td>
<td>Required</td>
<td>54.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ScheduledRecommendationInput

A scheduled custom recommendation.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>channel</td>
<td><a href="#">ConnectApi.RecommendationChannel</a></td>
<td>A way to tie custom recommendations together. For example, display recommendations in specific places in the UI or show recommendations based on time of day or geographic locations. Values are:</td>
<td>Optional for creating a scheduled recommendation</td>
<td>36.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- CustomChannel1—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels. For example, community managers can use Experience Builder to determine where recommendations appear.</td>
<td>If not specified, defaults to DefaultChannel. Don’t use when updating a scheduled recommendation</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- CustomChannel2—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- CustomChannel3—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- CustomChannel4—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- CustomChannel5—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- DefaultChannel—Default recommendation channel. Recommendations appear by default</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Property</td>
<td>Type</td>
<td>Description</td>
<td>Required or Optional</td>
<td>Available Version</td>
</tr>
<tr>
<td>--------------</td>
<td>---------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>on the Home and Question Detail pages of Customer Service and Partner Central Experience Builder templates. They also appear in the feed in the Salesforce mobile web and anywhere community managers add recommendations using Experience Builder. Use these channel values; you can’t rename or create other channels.</td>
<td>Optional</td>
<td>35.0</td>
</tr>
<tr>
<td>enabled</td>
<td>Boolean</td>
<td>Indicates whether scheduling is enabled. If true, the custom recommendation is enabled and appears in Experience Cloud sites. If false, custom recommendations in feeds in Salesforce mobile web aren’t removed, but no new custom recommendations appear. In Customer Service and Partner Central sites, disabled custom recommendations no longer appear.</td>
<td>Optional</td>
<td>35.0</td>
</tr>
<tr>
<td>rank</td>
<td>Integer</td>
<td>Relative rank of the scheduled custom recommendation indicated by ascending whole numbers starting with 1. Setting the rank is comparable to an insertion into an ordered list. The scheduled custom recommendation is inserted into the position specified by the rank. The rank of all the scheduled custom recommendations after it is pushed down. See Ranking scheduled custom recommendations example. If the specified rank is larger than the size of the list, the scheduled custom recommendation is put at the end of the list. The rank of the scheduled custom recommendation is the size of the list, instead of the one specified. If a rank is not specified, the scheduled custom recommendation is put at the end of the list.</td>
<td>Optional</td>
<td>35.0</td>
</tr>
<tr>
<td>recommendation AudienceId</td>
<td>String</td>
<td>ID of the audience for this scheduled custom recommendation. When updating a scheduled custom recommendation, specify ALL to remove the association between a</td>
<td>Optional</td>
<td>35.0</td>
</tr>
</tbody>
</table>
custom recommendation audience and a scheduled custom recommendation.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>recommendation</td>
<td>String</td>
<td>ID of the custom recommendation definition that this scheduled recommendation schedules.</td>
<td>Required to create a scheduled recommendation. You can't specify a recommendation DefinitionId when updating a scheduled recommendation.</td>
<td>35.0</td>
</tr>
<tr>
<td>DefinitionId</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Ranking scheduled custom recommendations example

If you have these scheduled custom recommendations:

<table>
<thead>
<tr>
<th>Scheduled Recommendations</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>ScheduledRecommendationA</td>
<td>1</td>
</tr>
<tr>
<td>ScheduledRecommendationB</td>
<td>2</td>
</tr>
<tr>
<td>ScheduledRecommendationC</td>
<td>3</td>
</tr>
</tbody>
</table>

And you include this information in the Scheduled Custom Recommendation Input:

<table>
<thead>
<tr>
<th>Scheduled Recommendation</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>ScheduledRecommendationD</td>
<td>2</td>
</tr>
</tbody>
</table>

The result is:

<table>
<thead>
<tr>
<th>Scheduled Recommendation</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>ScheduledRecommendationA</td>
<td>1</td>
</tr>
<tr>
<td>ScheduledRecommendationD</td>
<td>2</td>
</tr>
<tr>
<td>ScheduledRecommendationB</td>
<td>3</td>
</tr>
<tr>
<td>ScheduledRecommendationC</td>
<td>4</td>
</tr>
</tbody>
</table>

SEE ALSO:
createScheduledRecommendation(communityId, scheduledRecommendation)
**ConnectApi.SellerDetailsRequest**

Seller details for the tax calculation

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>code</td>
<td>String</td>
<td>Code used to identify the seller of the taxed items.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ServiceAppointmentInput**

Contains information about the service appointment.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>additionalInformation</td>
<td>String</td>
<td>The extra information about the service appointment.</td>
<td>Optional</td>
<td>53.0</td>
</tr>
<tr>
<td>appointmentType</td>
<td>String</td>
<td>The type of the appointment.</td>
<td>Optional</td>
<td>53.0</td>
</tr>
<tr>
<td>city</td>
<td>String</td>
<td>The name of the city.</td>
<td>Optional</td>
<td>53.0</td>
</tr>
<tr>
<td>comments</td>
<td>String</td>
<td>The comments about the appointment.</td>
<td>Optional</td>
<td>53.0</td>
</tr>
<tr>
<td>contactId</td>
<td>String</td>
<td>The ID of the contact associated with the parent record.</td>
<td>Optional</td>
<td>53.0</td>
</tr>
<tr>
<td>country</td>
<td>String</td>
<td>The name of the country.</td>
<td>Optional</td>
<td>53.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>The description of the appointment.</td>
<td>Optional</td>
<td>53.0</td>
</tr>
<tr>
<td>engagementChannelTypeId</td>
<td>String</td>
<td>ID of the engagement channel type to associate with the appointment.</td>
<td>Optional</td>
<td>56.0</td>
</tr>
</tbody>
</table>

You can use engagement channel type only if:

- **Schedule Appointments Using Engagement Channels** is enabled in Salesforce Scheduler Settings in your Salesforce org.
- Shifts are defined in the scheduling policy. For more information on setting up shifts in the scheduling policy, see Define Shift Rules in Scheduling Policy.

**Note:** Engagement channel types are not supported with operating hours rules in the scheduling policy.
### ConnectApi.ShiftsFromPatternInput

Shifts from a pattern.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>schedulingEndDate</td>
<td>String</td>
<td>Scheduling end date in YYYY-MM-DD format. Provide schedulingEndDate or schedulingOccurrences. Don't provide both.</td>
<td>Required if schedulingOccurrences isn't provided</td>
<td>51.0</td>
</tr>
<tr>
<td>schedulingOccurrences</td>
<td>Integer</td>
<td>Number of scheduling occurrences. Provide schedulingEndDate or schedulingOccurrences. Don't provide both.</td>
<td>Required if schedulingEndDate isn't provided</td>
<td>51.0</td>
</tr>
<tr>
<td>schedulingStartDate</td>
<td>String</td>
<td>Scheduling start date in YYYY-MM-DD format.</td>
<td>Required</td>
<td>51.0</td>
</tr>
<tr>
<td>Property</td>
<td>Type</td>
<td>Description</td>
<td>Required or Optional</td>
<td>Available Version</td>
</tr>
<tr>
<td>------------------------</td>
<td>------------</td>
<td>------------------------------------------------------------------------------</td>
<td>----------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>serviceResourceId</td>
<td>String</td>
<td>ID of the service resource to assign shifts to. In version 53.0 and later, use serviceResourceIdList.</td>
<td>Optional</td>
<td>51.0–52.0</td>
</tr>
<tr>
<td>serviceResourceId</td>
<td>List&lt;String&gt;</td>
<td>List of service resource IDs to assign shifts to.</td>
<td>Optional</td>
<td>53.0</td>
</tr>
<tr>
<td>serviceTerritoryId</td>
<td>String</td>
<td>ID of the service territory to assign shifts to.</td>
<td>Optional</td>
<td>51.0</td>
</tr>
<tr>
<td>shiftStatus</td>
<td>String</td>
<td>Status of the shifts. Default values are: Confirmed, Published, Tentative. Additional status values can be created.</td>
<td>Optional</td>
<td>52.0</td>
</tr>
</tbody>
</table>

**ConnectApi.SocialPostMassApprovalInput**

List of social post ids and the action to approve or reject publishing them.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isApproved</td>
<td>Boolean</td>
<td>Specifies whether to approve (true) or reject (false) publishing the social posts. If unspecified, defaults to false.</td>
<td>Optional</td>
<td>46.0</td>
</tr>
<tr>
<td>socialPostIdList</td>
<td>List&lt;String&gt;</td>
<td>A list of up to 200 social post IDs.</td>
<td>Required</td>
<td>46.0</td>
</tr>
</tbody>
</table>

**ConnectApi.StatusCapabilityInput**

Change the status of a feed post or comment.

This class is a subclass of ConnectApi.FeedElementCapabilityInput.
### Available Version

<table>
<thead>
<tr>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Required</td>
<td>37.0</td>
</tr>
</tbody>
</table>

### Description

- **ConnectApi.FeedEntityStatus**

  Status of the feed post or comment. Values are:

  - **Draft** — The feed post isn’t published but is visible to the author and users with Modify All Data or View All Data permission. Comments can’t be drafts.
  - **PendingReview** — The feed post or comment isn’t approved yet and therefore isn’t published or visible.
  - **Published** — The feed post or comment is approved and visible.

  Posts that have a status of PendingReview or Published can’t be changed to a status of Draft and vice versa.

### ConnectApi.StreamSubscriptionInput

An entity to subscribe to for a Chatter feed stream.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>entityId</td>
<td>String</td>
<td>The ID of any feed-enabled entity, such as a group, record, or user, that the context user can access. When subscribed, the entity’s feed is included in the feed stream.</td>
<td>Required</td>
<td>39.0</td>
</tr>
</tbody>
</table>

### ConnectApi.SurveyInvitationEmailInput

Survey invitation email.

### ConnectApi.Input Classes

**Apex Reference Guide**

SEE ALSO:

- ConnectApi.FeedElementCapabilitiesInput
- ConnectApi.ChatterStreamInput
- ConnectApi.SurveyInvitationEmailInput
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>allowGuestUserResponse</td>
<td>Boolean</td>
<td>Specifies whether participants who don’t have a Salesforce account can respond (true) or not (false).</td>
<td>Required</td>
<td>50.0</td>
</tr>
<tr>
<td>allowParticipantsAccessTheirResponse</td>
<td>Boolean</td>
<td>Specifies whether participants can see their responses (true) or not (false).</td>
<td>Required</td>
<td>50.0</td>
</tr>
<tr>
<td>associateRecordsWithRecipients</td>
<td>List&lt;ConnectApi.AssociateRecordsWithRecipientInput&gt;</td>
<td>Maps each recipient with another record that must be associated with the recipient’s survey invitation.</td>
<td>Optional</td>
<td>50.0</td>
</tr>
<tr>
<td>body</td>
<td>String</td>
<td>Content of the email. Specify the email body in case you don’t specify an email template. The email body must contain one of the following merge fields:</td>
<td>Optional</td>
<td>50.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• To embed a link to launch the survey: [[SURVEY_INVITATION_URL]]</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• To embed a survey question: {{{SurveyQuestion.QuestionName}}} and {{{SurveyQuestion.QuestionHtmlContent}}}</td>
<td></td>
<td></td>
</tr>
<tr>
<td>collectAnonymousResponse</td>
<td>Boolean</td>
<td>Specifies whether participants can respond anonymously (true) or not (false).</td>
<td>Required</td>
<td>50.0</td>
</tr>
<tr>
<td>communityId</td>
<td>String</td>
<td>ID of the site that’s used to open the survey for users outside your org.</td>
<td>Optional</td>
<td>50.0</td>
</tr>
<tr>
<td>emailTemplateId</td>
<td>String</td>
<td>ID of the Lightning email template that’s used to send the survey invitation. The template must contain the required merge fields that embed either the survey link or a question in the email. Only Lightning email templates are used to send survey invitations.</td>
<td>Optional</td>
<td>50.0</td>
</tr>
<tr>
<td>fromEmailAddress</td>
<td>String</td>
<td>Email ID of the user or the org-wide email address associated with the user’s profile.</td>
<td>Required</td>
<td>50.0</td>
</tr>
<tr>
<td>invitationExpirationDate</td>
<td>Datetime</td>
<td>Date on which the survey invitation expires.</td>
<td>Optional</td>
<td>50.0</td>
</tr>
<tr>
<td>invitationOwner</td>
<td>String</td>
<td>ID of the owner of the survey invitation records.</td>
<td>Optional</td>
<td>50.0</td>
</tr>
<tr>
<td>isPersonalInvitation</td>
<td>Boolean</td>
<td>Specifies whether an unique invitation is created for each participant (true) or not (false). When a participant responds using a personal invitation, the response record is associated with the participant's Salesforce record.</td>
<td>Required</td>
<td>50.0</td>
</tr>
<tr>
<td>Property</td>
<td>Type</td>
<td>Description</td>
<td>Required or Optional</td>
<td>Available Version</td>
</tr>
<tr>
<td>--------------------------</td>
<td>---------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>----------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>recipient Engagement Contexts</td>
<td>List&lt;ConnectApi.RecipientEngagementContextInput&gt;</td>
<td>Maps each recipient with the context based on which the survey invitation is emailed.</td>
<td>Optional</td>
<td>50.0</td>
</tr>
<tr>
<td>recipients</td>
<td>List&lt;String&gt;</td>
<td>List of up to 300 IDs of leads, contacts, or users to whom the survey invitation is emailed.</td>
<td>Required</td>
<td>50.0</td>
</tr>
<tr>
<td>shareInvitations With</td>
<td>List&lt;String&gt;</td>
<td>IDs of the users with whom the survey invitation records must be shared. The invitation records are shared with Read access.</td>
<td>Optional</td>
<td>50.0</td>
</tr>
<tr>
<td>subject</td>
<td>String</td>
<td>Subject of the email. Specify the subject in case you don’t specify an email template.</td>
<td>Optional</td>
<td>50.0</td>
</tr>
<tr>
<td>surveyQuestionIds</td>
<td>List&lt;String&gt;</td>
<td>IDs of the questions that are embedded in the email. You can send an email invitation for questions of the following types: Net Promoter Score (NPS), rating, and score.</td>
<td>Optional</td>
<td>50.0</td>
</tr>
</tbody>
</table>

**ConnectApi.TargetCollectionInput**

Collection of targets to create.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>targets</td>
<td>List&lt;ConnectApi.TargetInput&gt;</td>
<td>List of targets to create.</td>
<td>Required</td>
<td>48.0</td>
</tr>
</tbody>
</table>

**ConnectApi.TargetCollectionUpdateInput**

Collection of targets to update.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>targets</td>
<td>List&lt;ConnectApi.TargetUpdateInput&gt;</td>
<td>List of targets to update.</td>
<td>Required</td>
<td>48.0</td>
</tr>
</tbody>
</table>

**ConnectApi.TargetLocationInputRepresentation**

A set of inventory locations that together can fulfill an order.
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>locations</td>
<td>List&lt;ConnectApi.LocationInput&gt;</td>
<td>A list of locations with information about their country and postal codes.</td>
<td>Required</td>
<td>51.0</td>
</tr>
</tbody>
</table>

**ConnectApi.TargetInput**

Target to create.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>audienceId</td>
<td>String</td>
<td>ID of the audience to assign to the target.</td>
</tr>
<tr>
<td>groupName</td>
<td>String</td>
<td>Group name of the target. Groups bundle related target and audience pairs. You can have up to 2,000 groups and 500 targets per group. To determine the group name for targets of type ExperienceVariation, see Personalization Target Developer and Group Names in the Experience Cloud Developer Guide.</td>
</tr>
<tr>
<td>priority</td>
<td>Integer</td>
<td>Priority of the target. Within a group, priority determines which target is returned if the user matches more than one audience.</td>
</tr>
<tr>
<td>publishStatus</td>
<td>ConnectApi.PublishStatus</td>
<td>The publish status of the target. Values are: Draft, Live. We recommend setting the publish status to Draft. If you specify Live, your changes revert after you publish the site.</td>
</tr>
<tr>
<td>targetType</td>
<td>String</td>
<td>Type of target, indicating the nature of the data being targeted. Supported values include: ExperienceVariation (version 48.0 and later), Custom object API names, such as CustomObjectName__c (version 48.0 and later), NavigationLinkSet (version 49.0 and later), Topic (version 49.0 and later)</td>
</tr>
</tbody>
</table>
Available Version Required or Optional Description

- **CollaborationGroup** (version 49.0 and later)
- **KnowledgeArticle** (version 49.0 and later)
- **ContentDocument** (version 49.0 and later)
- **ManagedContent** (version 49.0 and later)
- **Report** (version 49.0 and later)
- **Dashboard** (version 49.0 and later)

You can have up to 2,500 ExperienceVariation targets and 25,000 record targets.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>targetValue</strong></td>
<td><strong>String</strong></td>
<td>Value of the target. If <code>targetType</code> is <code>ExperienceVariation</code>, <code>targetValue</code> is the developer name of the experience variation. If <code>targetType</code> is <code>CustomObjectName__c</code>, <code>targetValue</code> is the ID of the custom object. To determine the developer name for targets of type <code>ExperienceVariation</code>, see <a href="#">Personalization Target Developer and Group Names</a> in the Experience Cloud Developer Guide.</td>
<td>Required</td>
<td>48.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
[ConnectApi.TargetCollectionInput](#)

**ConnectApi.TargetUpdateInput**
Target to update.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>audienceId</strong></td>
<td><strong>String</strong></td>
<td>ID of the audience to assign to the target. Required if priority isn't specified. Otherwise, Optional</td>
<td>48.0</td>
<td></td>
</tr>
</tbody>
</table>
### Available Version

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>priority</td>
<td>Integer</td>
<td>Priority of the target. Within a group, priority determines which target is returned if the user matches more than one audience.</td>
<td>Required if <code>audienceId</code> isn't specified. Otherwise, Optional</td>
<td>48.0</td>
</tr>
<tr>
<td>targetId</td>
<td>String</td>
<td>ID of the target to update.</td>
<td>Required</td>
<td>48.0</td>
</tr>
</tbody>
</table>

#### ConnectApi.CalculateTaxRequest

Request to sent through the tax adapter to the external tax engine. Inputs with a `TaxTransactionType` of Debit represent a tax calculation request. Inputs with a `TaxTransactionType` of Credit represent a tax cancellation request.

Subclass of `ConnectApi.TaxTransactionRequest`.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isCommit</td>
<td>Boolean</td>
<td>Commits the transaction for tax calculation.</td>
<td>Required</td>
<td>55.0</td>
</tr>
<tr>
<td>taxEngineId</td>
<td>String</td>
<td>ID of the Salesforce tax engine entity used to represent the external tax engine.</td>
<td>Required</td>
<td>55.0</td>
</tr>
</tbody>
</table>
| taxTransactionType     | `ConnectApi.TaxTransaction.Type` | Type of tax transaction. Values are:  
• Credit—Transaction is a credit transaction.  
• Debit—Transaction is a debit transaction. | Required             | 55.0              |
| taxType                | `ConnectApi.CalculateTax.Type` | Type of tax calculation. Values are:  
• Actual—Calculated tax represents the final taxed amount for the transaction.  
• Estimated—Calculated tax represents only an estimated value before the transaction is finalized. | Required             | 55.0              |

#### ConnectApi.TaxAddressRequest

Address input representation for tax calculation.
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>city</td>
<td>String</td>
<td>City.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>country</td>
<td>String</td>
<td>Country.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>latitude</td>
<td>Double</td>
<td>Latitude.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>locationCode</td>
<td>String</td>
<td>Location code.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>longitude</td>
<td>Double</td>
<td>Longitude.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>postalCode</td>
<td>String</td>
<td>Postal code.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>state</td>
<td>String</td>
<td>State.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>street</td>
<td>String</td>
<td>Street.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
</tbody>
</table>

**ConnectApi.TaxAddressesRequest**

Addresses, including the Bill To address, Ship From address, Ship to address, and Sold To address.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>billTo</td>
<td>ConnectApi.TaxAddressRequest</td>
<td>Bill To address.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>shipFrom</td>
<td>ConnectApi.TaxAddressRequest</td>
<td>Ship From address.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>shipTo</td>
<td>ConnectApi.TaxAddressRequest</td>
<td>Ship To address.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>soldTo</td>
<td>ConnectApi.TaxAddressRequest</td>
<td>Sold To address.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
</tbody>
</table>

**ConnectApi.TaxCustomerDetailsRequest**

Customer details for the tax calculation.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>accountId</td>
<td>String</td>
<td>ID of the customer's account.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>code</td>
<td>String</td>
<td>Customer code.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>exemptionNo</td>
<td>String</td>
<td>Tax exemption number.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
</tbody>
</table>
### ConnectApi.TaxLineItemRequest

A list of line items passed to the tax engine for tax calculation.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>exemptionReason</td>
<td>String</td>
<td>Tax exemption reason.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>addresses</td>
<td>ConnectApi.TaxAddressesRequest</td>
<td>Addresses, including the Bill To address, Ship From address, Ship To address, and Sold To address.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>amount</td>
<td>Double</td>
<td>Amount of the line item.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the line item.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>effectiveDate</td>
<td>Datetime</td>
<td>Date to apply the tax calculation to the line item.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>lineNumber</td>
<td>String</td>
<td>Line number of the line item.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>productCode</td>
<td>String</td>
<td>Product code of the line item.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>quantity</td>
<td>Double</td>
<td>Quantity of the line item.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>taxCode</td>
<td>String</td>
<td>Tax code for the line item.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
</tbody>
</table>

### ConnectApi.TaxTransactionRequest

Information about the tax transaction sent to the tax adapter as part of a tax calculation request.

This class is abstract.

Superclass of `ConnectApi.CalculateTaxRequest`.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>addresses</td>
<td>ConnectApi.TaxAddressesRequest</td>
<td>Addresses, including the Bill To address, Ship From address, Ship To address, and Sold To address.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Three-letter ISO 4217 currency code associated with the payment group record.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>customerDetails</td>
<td>ConnectApi.TaxCustomerDetailsRequest</td>
<td>Customer details for the tax calculation.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>Property</td>
<td>Type</td>
<td>Description</td>
<td>Required or Optional</td>
<td>Available Version</td>
</tr>
<tr>
<td>------------------</td>
<td>---------------</td>
<td>------------------------------------------------------------------------------</td>
<td>----------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Information about whether the tax transaction failed or was successful.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>documentCode</td>
<td>String</td>
<td>Document code.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>effectiveDate</td>
<td>Datetime</td>
<td>The date that tax is applied to the taxed entity.</td>
<td>Required</td>
<td>55.0</td>
</tr>
<tr>
<td>lineItems</td>
<td>List&lt;ConnectApi.TaxLineItemRequest&gt;</td>
<td>The line items on which tax was calculated.</td>
<td>Required</td>
<td>55.0</td>
</tr>
<tr>
<td>referenceDocumentCode</td>
<td>String</td>
<td>The original document code. Used in case of subsequent transactions such as credit tax.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>referenceEntityId</td>
<td>String</td>
<td>ID of the reference entity used during tax calculation.</td>
<td>Optional</td>
<td>55.0</td>
</tr>
<tr>
<td>transactionDate</td>
<td>Datetime</td>
<td>The date that the tax transaction occurred.</td>
<td>Optional</td>
<td>53.0</td>
</tr>
</tbody>
</table>

**ConnectApi.TextSegmentInput**

Include a text segment in a feed item or comment.
Subclass of ConnectApi.MessageSegmentInput.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>text</td>
<td>String</td>
<td>Plain text for this segment. If hashtags or links are detected in text, they're included in the comment as hashtag and link segments. Mentions aren't detected in text and aren't separated out of the text. Mentions require ConnectApi.MentionSegmentInput.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- Edit a Comment
- Edit a Feed Element
- Edit a Question Title and Post
- Post a Rich-Text Feed Element with Inline Image
- ConnectApi.MessageBodyInput

**ConnectApi.TopicInput**

Update a topic's name or description or merge topics.
### ConnectApi.TopicNamesInput

A list of topic names to replace currently assigned topics. Also a list of suggested topics to assign.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>topicNames</td>
<td>List&lt;String&gt;</td>
<td>A list of up to 10 topic names for a feed item or 100 topic names for a record.</td>
<td>Required</td>
<td>35.0</td>
</tr>
<tr>
<td>topicSuggestions</td>
<td>List&lt;String&gt;</td>
<td>A list of suggested topics to assign to a record or feed item to improve future topic suggestions.</td>
<td>Optional</td>
<td>37.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- `updateTopic(communityId, topicId, topic)`
- `reassignTopicsByName(communityId, recordId, topicNames)`
- `ConnectApi.ArticleTopicAssignmentJobInput`

### ConnectApi.TopicsCapabilityInput

Assign topics to a feed element.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>contextTopic</td>
<td>String</td>
<td>Name of the parent topic in the site to which the feed element belongs.</td>
<td>Optional</td>
<td>38.0</td>
</tr>
</tbody>
</table>
### ConnectApi.FeedElementCapabilitiesInput

List of topics to assign to the feed element.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>topics</td>
<td>List&lt;String&gt;</td>
<td>List of topics to assign to the feed element.</td>
<td>Required</td>
<td>38.0</td>
</tr>
</tbody>
</table>

See Also:

ConnectApi.FeedElementCapabilitiesInput

### ConnectApi.UpDownVoteCapabilityInput

Upvote or downvote a feed element or a comment.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>vote</td>
<td>ConnectApi.UpDownVoteValue</td>
<td>Type of vote for a feed element or comment. Values are: Down, None, Up.</td>
<td>Required</td>
<td>41.0</td>
</tr>
</tbody>
</table>

### ConnectApi.UpdateServiceAppointmentInput

Contains information to update a service appointment.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>serviceAppointmentId</td>
<td>String</td>
<td>The ID of the service appointment to be modified.</td>
<td>Required</td>
<td>53.0</td>
</tr>
</tbody>
</table>
| assignedResources   | List<ConnectApi.AssignedResourcesInput> | Represents the service resources who are assigned to a service appointment. When updating an appointment, pass the complete list of required resources. If you don't pass a resource who is already assigned to the appointment, the API deletes that assigned resource. For example, suppose that an existing service appointment has assigned resources: A and B and you pass B and C in assigned resources in the PATCH request. The API checks the resource availability of B and C for existing work type and service territory, and if both are available, the service appointment gets updated with:
• Resource A—Deleted | Optional             | 53.0 |
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>lead</td>
<td>ConnectApi.LeadInput</td>
<td>Represents a prospect or lead.</td>
<td>Required if serviceAppointment isn't provided.</td>
<td>53.0</td>
</tr>
<tr>
<td>schedulingPolicyId</td>
<td>String</td>
<td>The ID of the AppointmentSchedulingPolicy object. If no scheduling policy is passed in the request body, the default configurations are used. The only scheduling policy configuration that is used in determining time slots is the enforcement of account visiting hours.</td>
<td>Optional</td>
<td>53.0</td>
</tr>
<tr>
<td>serviceAppointment</td>
<td>ConnectApi.ServiceAppointmentInput</td>
<td>Represents the service appointment details to book an appointment. When updating an appointment, pass only the fields that must be updated.</td>
<td>Required if lead isn't provided.</td>
<td>53.0</td>
</tr>
</tbody>
</table>

**ConnectApi.UserInput**

Update a user’s About Me information.
### ConnectApi.UserDetail.output

The `aboutMe` property of a `ConnectApi.UserDetail` output object. This property populates the About Me section of the user profile, which is visible to all members of an Experience Cloud site or org.

**Available Version**: 29.0

**Description**: String

**SEE ALSO**: `updateUser(communityId, userId, userInput)`

---

### ConnectApi.WishlistInput

Create a wishlist.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the wishlist.</td>
<td>Required</td>
<td>49.0</td>
</tr>
<tr>
<td>products</td>
<td>List&lt;ConnectApi.WishlistItemInput&gt;</td>
<td>List of products to add to the wishlist.</td>
<td>Optional</td>
<td>49.0</td>
</tr>
</tbody>
</table>

**SEE ALSO**: `ConnectApi.WishlistInput`, `ConnectApi.WishlistUpdateInput`

---

### ConnectApi.WishlistItemInput

Item to update or add to a wishlist.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>productId</td>
<td>String</td>
<td>ID of the product to update or add to the wishlist.</td>
<td>Required</td>
<td>49.0</td>
</tr>
</tbody>
</table>

**SEE ALSO**: `ConnectApi.WishlistInput`

---

### ConnectApi.WishlistUpdateInput

Update a wishlist name.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>Wishlist name to update.</td>
<td>Required</td>
<td>50.0</td>
</tr>
</tbody>
</table>

---

### Retired ConnectApi Input Classes

These `ConnectApi` input classes are retired.
IN THIS SECTION:

- **ConnectApi.CanvasAttachmentInput**
  Used to attach a canvas app to a feed item.

- **ConnectApi.ContentAttachmentInput**
  Used to attach existing content to a comment or feed item.

- **ConnectApi.DatacloudOrderInput**
  Input representation for a Datacloud order to purchase contacts or companies and retrieve purchase information.

- **ConnectApi.FeedItemAttachmentInput**
  Used to attach a file to a feed item.

- **ConnectApi.LinkAttachmentInput**
  Add links to a feed item.

- **ConnectApi.NewFileAttachmentInput**
  Attach a new file to a feed item.

- **ConnectApi.PollAttachmentInput**
  Attach a poll to a feed item.

**ConnectApi.CanvasAttachmentInput**

Used to attach a canvas app to a feed item.

**Important:** This class isn’t available in version 32.0 and later. In version 32.0 and later, use **ConnectApi.CanvasCapabilityInput**.

Subclass of **ConnectApi.FeedItemAttachmentInput**.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>String</td>
<td>Optional. The description of the canvas app.</td>
<td>29.0–31.0</td>
</tr>
<tr>
<td>developerName</td>
<td>String</td>
<td>The developer name (API name) of the canvas app</td>
<td>29.0–31.0</td>
</tr>
<tr>
<td>height</td>
<td>String</td>
<td>Optional. The height of the canvas app in pixels. Default height is 200 pixels.</td>
<td>29.0–31.0</td>
</tr>
<tr>
<td>namespacePrefix</td>
<td>String</td>
<td>Optional. The namespace prefix of the Developer Edition organization in which the canvas app was created.</td>
<td>29.0–31.0</td>
</tr>
<tr>
<td>parameters</td>
<td>String</td>
<td>Optional. Parameters passed to the canvas app in JSON format. Example:</td>
<td>29.0–31.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>{&quot;isUpdated&quot;:&quot;true&quot;}</td>
<td></td>
</tr>
<tr>
<td>thumbnailUrl</td>
<td>String</td>
<td>Optional. A URL to a thumbnail image for the canvas app. Maximum dimensions are 120x120 pixels.</td>
<td>29.0–31.0</td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>The title of the link used to call the canvas app.</td>
<td>29.0–31.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ContentAttachmentInput**

Used to attach existing content to a comment or feed item.
**Important:** This class isn’t available in version 32.0 and later. In version 32.0 and later, use `ConnectApi.ContentCapabilityInput`.

Subclass of `ConnectApi.FeedItemAttachmentInput`.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>contentDocumentId</td>
<td>String</td>
<td>ID of the existing content.</td>
<td>28.0–31.0</td>
</tr>
</tbody>
</table>

**ConnectApi.DatacloudOrderInput**

Input representation for a Datacloud order to purchase contacts or companies and retrieve purchase information.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Required or Optional</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>companyIds</td>
<td>String</td>
<td>A comma-separated list of identification numbers for the companies to be purchased. You can’t include any contact IDs or your purchase fails.</td>
<td>Required to purchase companies</td>
<td>32.0</td>
</tr>
<tr>
<td>contactIds</td>
<td>String</td>
<td>A comma-separated list of identification numbers for the contacts to be purchased. You can’t include any company IDs or your purchase fails.</td>
<td>Required to purchase contacts</td>
<td>32.0</td>
</tr>
<tr>
<td>userType</td>
<td>ConnectDatacloudUserTypeEnum</td>
<td>Indicates the Data.com user type to be used. There are two user types. • Monthly (default) • Listpool</td>
<td>Optional</td>
<td>32.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

`postOrder(orderInput)`

**ConnectApi.FeedItemAttachmentInput**

Used to attach a file to a feed item.

**Important:** This class isn’t available in version 32.0 and later. In version 32.0 and later, use `ConnectApi.FeedElementCapabilityInput`.

This class is abstract and has no public constructor. You can make an instance only of a subclass.

Superclass for:

- `ConnectApi.CanvasAttachmentInput`
- `ConnectApi.ContentAttachmentInput`
- `ConnectApi.LinkAttachmentInput`
ConnectApi.LinkAttachmentInput

Add links to a feed item.

**Important:** This class isn’t available in version 32.0 and later. In version 32.0 and later, use ConnectApi.LinkCapabilityInput.

Subclass of ConnectApi.FeedItemAttachmentInput.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>String</td>
<td>URL to be used for the link</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>urlName</td>
<td>String</td>
<td>Title of the link</td>
<td>28.0–31.0</td>
</tr>
</tbody>
</table>

ConnectApi.NewFileAttachmentInput

Attach a new file to a feed item.

**Important:** This class isn’t available in version 32.0 and later. In version 32.0 and later, use ConnectApi.ContentCapabilityInput.

The actual binary file, that is the attachment, is provided as part of the BinaryInput in the method that takes this attachment input, such as postFeedItem or postComment.

Subclass of ConnectApi.FeedItemAttachmentInput.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the file to be uploaded.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>The title of the file. This value is required and is also used as the file name. For example, if the title is My Title, and the file is a .txt file, the file name is My Title.txt.</td>
<td>28.0–31.0</td>
</tr>
</tbody>
</table>

ConnectApi.PollAttachmentInput

Attach a poll to a feed item.

**Important:** This class isn’t available in version 32.0 and later. In version 32.0 and later, use ConnectApi.PollCapabilityInput.

Subclass of ConnectApi.FeedItemAttachmentInput.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>pollChoices</td>
<td>List&lt;String&gt;</td>
<td>The text labels for the poll items. Polls must contain between 2 to 10 poll choices.</td>
<td>28.0–31.0</td>
</tr>
</tbody>
</table>

ConnectApi Output Classes

Most ConnectApi methods return instances of ConnectApi output classes.
All properties are read-only, except for instances of output classes created within test code.
All output classes are concrete unless marked abstract in this documentation.
All concrete output classes have no-argument constructors that you can invoke only from test code. See Testing ConnectApi Code.

**ConnectApi.AbstractCartItem**

A cart item.
This class is abstract.
Superclass of:
- ConnectApi.CartItem
- ConnectApi.CartItemWithoutPrice

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>billingFrequency</td>
<td>ConnectApi.BillingFrequency</td>
<td>Reserved for future use.</td>
<td>59.0</td>
</tr>
<tr>
<td>cartId</td>
<td>String</td>
<td>ID of the cart.</td>
<td>49.0</td>
</tr>
<tr>
<td>cartItemId</td>
<td>String</td>
<td>ID of the item.</td>
<td>49.0</td>
</tr>
<tr>
<td>messagesSummary</td>
<td>ConnectApi.CartMessagesSummary</td>
<td>Messages summary for the item.</td>
<td>49.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the item.</td>
<td></td>
</tr>
<tr>
<td>productDetails</td>
<td>ConnectApi.CartItemProduct</td>
<td>Summary of the product details.</td>
<td>49.0</td>
</tr>
<tr>
<td>productId</td>
<td>String</td>
<td>ID of the product.</td>
<td>49.0</td>
</tr>
<tr>
<td>productSellingModelId</td>
<td>String</td>
<td>Reserved for future use.</td>
<td>59.0</td>
</tr>
<tr>
<td>quantity</td>
<td>String</td>
<td>Quantity of the item.</td>
<td>49.0</td>
</tr>
<tr>
<td>subscriptionTerm</td>
<td>Integer on page 3125</td>
<td>Reserved for future use.</td>
<td>59.0</td>
</tr>
<tr>
<td>type</td>
<td>ConnectApi.CartItemType</td>
<td>Type of item in a cart. Values are:</td>
<td>49.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• DeliveryCharge</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Product</td>
<td></td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.CartItemResult

**ConnectApi.AbstractContentHubItemType**

An item type associated with a repository folder.
This class is abstract.
Superclass of:
ConnectApi.ContentHubItemTypeDetail
ConnectApi.ContentHubItemTypeSummary

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>contentStreamSupport</td>
<td>ConnectApi.ContentHubStreamSupport</td>
<td>Support for content streaming. Values are:</td>
<td>39.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ContentStreamAllowed</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ContentStreamNotAllowed</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ContentStreamRequired</td>
<td></td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the item type.</td>
<td>39.0</td>
</tr>
<tr>
<td>displayName</td>
<td>String</td>
<td>Display name of the item type.</td>
<td>39.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the item type.</td>
<td>39.0</td>
</tr>
<tr>
<td>isVersionable</td>
<td>Boolean</td>
<td>Indicates whether the item type can have versions.</td>
<td>39.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>URL to the detailed information of the item type.</td>
<td>39.0</td>
</tr>
</tbody>
</table>

ConnectApi.AbstractDirectoryEntrySummary

A directory entry with summary information.
This class is abstract.
Superclass of:
• ConnectApi.RepositoryGroupSummary
• ConnectApi.RepositoryUserSummary

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>domain</td>
<td>String</td>
<td>Domain of the directory entry.</td>
<td>39.0</td>
</tr>
<tr>
<td>email</td>
<td>String</td>
<td>Email of the directory entry.</td>
<td>39.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the directory entry.</td>
<td>39.0</td>
</tr>
<tr>
<td>type</td>
<td>ConnectApi.ContentHubDirectoryEntryType</td>
<td>Type of directory entry. Values are:</td>
<td>39.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• GroupEntry</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• UserEntry</td>
<td></td>
</tr>
</tbody>
</table>

ConnectApi.AbstractExtensionInformation

Extension information.
This class is abstract.
Superclass of ConnectApi.LightningExtensionInformation.
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>extension</td>
<td><code>extensionInformationType</code></td>
<td>Information type of the extension. Values are:</td>
<td>40.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Lightning</td>
<td></td>
</tr>
</tbody>
</table>

**ConnectApi.AbstractGatewayCommonResponse**

Payment gateway response fields commonly used in payment services.

This class is abstract.


<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>gatewayAvsCode</td>
<td><code>String</code></td>
<td>Used to verify the address mapped to a payment method when the payments platform requests tokenization from the payment gateway.</td>
<td>50.0</td>
</tr>
<tr>
<td>gatewayDate</td>
<td><code>Datetime</code></td>
<td>Date when the notification occurred. Some gateways don’t send this value.</td>
<td>50.0</td>
</tr>
<tr>
<td>gatewayMessage</td>
<td><code>String</code></td>
<td>Error messages that the gateway returned for the notification request. Maximum length of 255 characters.</td>
<td>50.0</td>
</tr>
<tr>
<td>gatewayResultCode</td>
<td><code>String</code></td>
<td>Gateway-specific result code. You can map the result code to a Salesforce-specific result code. Maximum length of 64 characters.</td>
<td>50.0</td>
</tr>
<tr>
<td>gatewayResultCodeDescription</td>
<td><code>String</code></td>
<td>A description of the gateway-specific result code that a payment gateway returned. Maximum length of 1,000 characters.</td>
<td>50.0</td>
</tr>
<tr>
<td>salesforceResultCode</td>
<td><code>String</code></td>
<td>The Salesforce result code for the gateway result code.</td>
<td>50.0</td>
</tr>
</tbody>
</table>

**ConnectApi.AbstractGatewayResponse**

Payment gateway response fields used in sale, authorization, and capture services.

This class is abstract.

Subclass of `ConnectApi.AbstractGatewayCommonResponse`.

Superclass of:
- `ConnectApi.AuthReversalGatewayResponse`
- `ConnectApi.AuthorizationGatewayResponse`
- `ConnectApi.AuthorizationReversalResponse`
- `ConnectApi.CaptureGatewayResponse`
- `ConnectApi.PaymentMethodTokenizationGatewayResponse`
ConnectApi.PostAuthGatewayResponse
ConnectApi.RefundGatewayResponse
ConnectApi.SaleGatewayResponse

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>gatewayReferenceDetails</td>
<td>String</td>
<td>Provides information about the gateway communication.</td>
<td>50.0</td>
</tr>
<tr>
<td>gatewayReferenceNumber</td>
<td>String</td>
<td>Unique transaction ID created by the payment gateway.</td>
<td>50.0</td>
</tr>
</tbody>
</table>

ConnectApi.AbstractManagedContentDeliveryDocument

Managed content delivery document.
This class is abstract.
Superclass of:
- ConnectApi.ManagedContentDeliveryDocument
- ConnectApi.ManagedContentDeliveryDocumentSummary

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>contentKey</td>
<td>String</td>
<td>Globally unique identifier (GUID) for the managed content.</td>
<td>55.0</td>
</tr>
<tr>
<td>contentType</td>
<td>ConnectApi.ManagedContentTypeSummary</td>
<td>Type of managed content.</td>
<td>55.0</td>
</tr>
<tr>
<td>language</td>
<td>String</td>
<td>Language locale of the managed content.</td>
<td>55.0</td>
</tr>
<tr>
<td>managedContentId</td>
<td>String</td>
<td>ID of the managed content.</td>
<td>55.0</td>
</tr>
<tr>
<td>publishedDate</td>
<td>Datetime</td>
<td>Most recent publish date of the managed content.</td>
<td>55.0</td>
</tr>
<tr>
<td>resourceUrl</td>
<td>String</td>
<td>URL to the single content delivery resource.</td>
<td>55.0</td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>Title of the managed content.</td>
<td>55.0</td>
</tr>
<tr>
<td>unauthenticatedUrl</td>
<td>String</td>
<td>Public URL for the managed content.</td>
<td>55.0</td>
</tr>
<tr>
<td>urlName</td>
<td>String</td>
<td>URL name of the managed content.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.ManagedContentDeliveryDocumentCollection

ConnectApi.AbstractManagedContentReference

Managed content reference.
This class is abstract.
Superclass of:
- ConnectApi.ManagedContentReference
- ConnectApi.ManagedContentReferenceSummary

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>contentKey</td>
<td>String</td>
<td>Unique identifier for the managed content reference.</td>
<td>54.0</td>
</tr>
<tr>
<td>managedContentId</td>
<td>String</td>
<td>ID of the managed content reference.</td>
<td>54.0</td>
</tr>
<tr>
<td>resourceUrl</td>
<td>String</td>
<td>URL to the single content delivery resource.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.ManagedContentDeliveryDocumentCollection

**ConnectApi.AbstractMessageBody**

Abstract message body.
This class is abstract.
Superclass of:
- ConnectApi.FeedBody
- ConnectApi.MessageBody

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isRichText</td>
<td>Boolean</td>
<td>Indicates whether the body is rich text.</td>
<td>35.0</td>
</tr>
<tr>
<td>messageSegments</td>
<td>List&lt;ConnectApi.MessageSegment&gt;</td>
<td>List of message segments</td>
<td>28.0</td>
</tr>
<tr>
<td>text</td>
<td>String</td>
<td>Display-ready text. Use this text if you don't want to process the message segments.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

**ConnectApi.AbstractNBAAction**

A recommended action of recommendation strategy.
This class is abstract.
Superclass of ConnectApi.NBAFlowAction.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>parameters</td>
<td>List&lt;ConnectApi.NBAActionParameter&gt;</td>
<td>List of parameters to pass to the action.</td>
<td>45.0</td>
</tr>
</tbody>
</table>
**Available Version**

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
</table>
| type          | ConnectApi.NBAActionType | Type of action. Values are:  
- Flow—Automated process tool with multiple subtypes. | 45.0 |

SEE ALSO:

ConnectApi.NBARecommendation

**ConnectApi.AbstractNBATarget**

A recommendation target of a recommendation strategy.

This class is abstract.

Superclass of ConnectApi.NBANativeRecommendation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
</table>
| type          | ConnectApi.NBATargetType | Type of target. Values are:  
- Recommendation | 45.0 |

SEE ALSO:

ConnectApi.NBARecommendation

**ConnectApi.AbstractRecommendation**

A Chatter, custom, or static recommendation.

This class is abstract.

Superclass of:

- ConnectApi.EntityRecommendation
- ConnectApi.NonEntityRecommendation

ConnectApi.NonEntityRecommendation isn’t used in version 34.0 and later. In version 34.0 and later, ConnectApi.EntityRecommendation is used for all recommendations.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>explanation</td>
<td>ConnectApi.RecommendationExplanation</td>
<td>The Chatter, custom, or static recommendation explanation.</td>
<td>32.0</td>
</tr>
<tr>
<td>platformActionGroup</td>
<td>ConnectApi.PlatformActionGroup</td>
<td>A platform action group instance with state appropriate for the context user.</td>
<td>34.0</td>
</tr>
</tbody>
</table>
### ConnectApi.AbstractRecommendationExplanation

Explanation for a Chatter recommendation.

This class is abstract.

Superclass of `ConnectApi.RecommendationExplanation`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>summary</td>
<td>String</td>
<td>Summary explanation for the Chatter recommendation.</td>
<td>32.0</td>
</tr>
<tr>
<td>type</td>
<td><code>ConnectApi.RecommendationExplanationType</code></td>
<td>Indicates the reason for the Chatter recommendation.</td>
<td>32.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ArticleHasRelatedContent—Articles with related content to a context article.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ArticleViewedTogether—Articles often viewed together with the article that the context user just viewed.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ArticleViewedTogetherWithViewers—Articles often viewed together with other records that the context user views.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Custom—Custom recommendations.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FilePopular—Files with many followers or views.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FileViewedTogether—Files often viewed at the same time as other files that the context user views.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FollowedTogetherWithFollowees—Users often followed together with other records that the context user follows.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• GroupMembersFollowed—Groups with members that the context user follows.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• GroupNew—Recently created groups.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• GroupPopular—Groups with many active members.</td>
<td></td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>-------------------------------------------------------------</td>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>ItemViewedTogether</td>
<td></td>
<td>Records often viewed at the same time as other records that the context user views.</td>
<td></td>
</tr>
<tr>
<td>PopularApp</td>
<td></td>
<td>Applications that are popular.</td>
<td></td>
</tr>
<tr>
<td>RecordOwned</td>
<td></td>
<td>Records that the context user owns.</td>
<td></td>
</tr>
<tr>
<td>RecordParentOfFollowed</td>
<td></td>
<td>Parent records of records that the context user follows.</td>
<td></td>
</tr>
<tr>
<td>RecordViewed</td>
<td></td>
<td>Records that the context user recently viewed.</td>
<td></td>
</tr>
<tr>
<td>TopicFollowedTogether</td>
<td></td>
<td>Topics often followed together with the record that the context user just followed.</td>
<td></td>
</tr>
<tr>
<td>TopicFollowedTogetherWithFollowees</td>
<td></td>
<td>Topics often followed together with other records that the context user follows.</td>
<td></td>
</tr>
<tr>
<td>TopicPopularFollowed</td>
<td></td>
<td>Topics with many followers.</td>
<td></td>
</tr>
<tr>
<td>TopicPopular Liked</td>
<td></td>
<td>Topics on posts that have many likes.</td>
<td></td>
</tr>
<tr>
<td>UserDirectReport</td>
<td></td>
<td>Users who report to the context user.</td>
<td></td>
</tr>
<tr>
<td>UserFollowedTogether</td>
<td></td>
<td>Users often followed together with the record that the context user followed.</td>
<td></td>
</tr>
<tr>
<td>UserFollowsSameUsers</td>
<td></td>
<td>Users who follow the same users as the context user.</td>
<td></td>
</tr>
<tr>
<td>UserManager</td>
<td></td>
<td>The context user's manager.</td>
<td></td>
</tr>
<tr>
<td>UserNew</td>
<td></td>
<td>Recently created users.</td>
<td></td>
</tr>
<tr>
<td>UserPeer</td>
<td></td>
<td>Users who report to the same manager as the context user.</td>
<td></td>
</tr>
<tr>
<td>UserPopular</td>
<td></td>
<td>Users with many followers.</td>
<td></td>
</tr>
<tr>
<td>UserViewingSameRecords</td>
<td></td>
<td>Users who view the same records as the context user.</td>
<td></td>
</tr>
</tbody>
</table>

**ConnectApi.AbstractRecordField**

A field on a record.

This class is abstract.

Superclass of:

- ConnectApi.BlankRecordField
ConnectApi.LabeledRecordField

Message segments in a feed item are typed as ConnectApi.MessageSegment. Feed item capabilities are typed as ConnectApi.FeedItemCapability. Record fields are typed as ConnectApi.AbstractRecordField. These classes are all abstract and have several concrete subclasses. At runtime you can use instanceof to check the concrete types of these objects and then safely proceed with the corresponding downcast. When you downcast, you must have a default case that handles unknown subclasses.

Important: The composition of a feed can change between releases. Write your code to handle instances of unknown subclasses.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>String</td>
<td>Type of the field. One of these values:</td>
<td>29.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Address</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Blank</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Boolean</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Compound</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• CreatedBy</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Date</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• DateTime</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Email</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• LastModifiedBy</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Location</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Number</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Percent</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Phone</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Picklist</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Reference</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Text</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Time</td>
<td></td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.RecordViewSection

ConnectApi.AbstractRecordView

A view of any record in the org, including a custom object record. This object is used if a specialized object, such as User or ChatterGroup, isn't available for the record type.
This class is abstract.
Subclass of ConnectApi.ActorWithId.
Superclass of:
• ConnectApi.RecordSummary
ConnectApi.RecordView

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>The localized name of the record.</td>
<td>29.0</td>
</tr>
</tbody>
</table>

ConnectApi.AbstractRepositoryFile

A repository file.
This class is abstract.
Subclass of ConnectApi.AbstractRepositoryItem.
Superclass of:
- ConnectApi.RepositoryFileDetail
- ConnectApi.RepositoryFileSummary

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>checkinComment</td>
<td>String</td>
<td>Checkin comment of the file.</td>
<td>39.0</td>
</tr>
<tr>
<td>contentBody</td>
<td>String</td>
<td>Text of the file's content if available, otherwise null.</td>
<td>43.0</td>
</tr>
<tr>
<td>contentSize</td>
<td>Integer</td>
<td>Length in bytes of the content of the file.</td>
<td>39.0</td>
</tr>
<tr>
<td>downloadUrl</td>
<td>String</td>
<td>URL to the repository file content.</td>
<td>39.0</td>
</tr>
<tr>
<td>externalContentUrl</td>
<td>String</td>
<td>URL of this file's content in the external system.</td>
<td>39.0</td>
</tr>
<tr>
<td>externalDocumentUrl</td>
<td>String</td>
<td>URL of this file in the external system.</td>
<td>39.0</td>
</tr>
<tr>
<td>externalFilePermissionInformation</td>
<td>ConnectApi.ExternalFilePermissionInformation</td>
<td>External file permission information, such as available groups, available permission types, and current sharing status, or null if includeExternalFilePermissionsInfo is false.</td>
<td>39.0</td>
</tr>
<tr>
<td>mimeType</td>
<td>String</td>
<td>Mime type of the file.</td>
<td>39.0</td>
</tr>
<tr>
<td>previewUrlThumbnail</td>
<td>String</td>
<td>URL to the thumbnail preview (240 x 180 PNG).</td>
<td>39.0</td>
</tr>
<tr>
<td>previewUrlThumbnailBig</td>
<td>String</td>
<td>URL to the big thumbnail preview (720 x 480 PNG).</td>
<td>39.0</td>
</tr>
<tr>
<td>previewUrlThumbnailTiny</td>
<td>String</td>
<td>URL to the tiny thumbnail preview (120 x 90 PNG).</td>
<td>39.0</td>
</tr>
<tr>
<td>previewsUrl</td>
<td>String</td>
<td>URL to the previews.</td>
<td>39.0</td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>Title of the file.</td>
<td>39.0</td>
</tr>
<tr>
<td>versionId</td>
<td>String</td>
<td>ID of the file version in the external system.</td>
<td>39.0</td>
</tr>
</tbody>
</table>
**ConnectApi.AbstractRepositoryFolder**

A repository folder.

This class is abstract.

Subclass of **ConnectApi.AbstractRepositoryItem**.

Superclass of:
- **ConnectApi.RepositoryFolderDetail**
- **ConnectApi.RepositoryFolderSummary**

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>externalFolderUrl</td>
<td>String</td>
<td>URL of this folder in the external system.</td>
<td>39.0</td>
</tr>
<tr>
<td>folderItemsUrl</td>
<td>String</td>
<td>URL that lists the files and folders in this folder.</td>
<td>39.0</td>
</tr>
<tr>
<td>path</td>
<td>String</td>
<td>Absolute path of the folder in the external system.</td>
<td>39.0</td>
</tr>
</tbody>
</table>

**ConnectApi.AbstractRepositoryItem**

A repository item.

This class is abstract.

Superclass of:
- **ConnectApi.AbstractRepositoryFile**
- **ConnectApi.AbstractRepositoryFolder**

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>createdBy</td>
<td>String</td>
<td>Name of the user who created the item.</td>
<td>39.0</td>
</tr>
<tr>
<td>createdDate</td>
<td>Datetime</td>
<td>Date the item was created.</td>
<td>39.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the item.</td>
<td>39.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the item.</td>
<td>39.0</td>
</tr>
<tr>
<td>itemTypeUrl</td>
<td>String</td>
<td>URL to the item type information.</td>
<td>39.0</td>
</tr>
<tr>
<td>modifiedBy</td>
<td>String</td>
<td>Name of the user who last modified the item.</td>
<td>39.0</td>
</tr>
<tr>
<td>modifiedDate</td>
<td>Datetime</td>
<td>Date the item was last modified.</td>
<td>39.0</td>
</tr>
<tr>
<td>motif</td>
<td>ConnectApi.Motif</td>
<td>Motif of the item.</td>
<td>39.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the item.</td>
<td>39.0</td>
</tr>
<tr>
<td>repository</td>
<td>ConnectApi.Reference</td>
<td>Item external repository.</td>
<td>39.0</td>
</tr>
<tr>
<td>type</td>
<td>String</td>
<td>Item type, file or folder.</td>
<td>39.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>The URL to the item.</td>
<td>39.0</td>
</tr>
</tbody>
</table>
ConnectApi.AbstractUserMissionActivity

User activity associated with missions.

This class is abstract.

Superclass of:

- ConnectApi.UserMission
- ConnectApi.UserMissionActivity

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>activityCount</td>
<td>Integer</td>
<td>Number of mission activities of the specified type for the user.</td>
<td>45.0</td>
</tr>
<tr>
<td>activityType</td>
<td>String</td>
<td>Type of mission activity for a user. Values are:</td>
<td>45.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FeedItemAnswerAQuestion—User answered a question.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FeedItemLikeSomething—User liked a post or comment.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FeedItemMarkAnswerAsBest—User marked an answer as the best answer.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FeedItemPostQuestion—User posted a question.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FeedItemReceiveAComment—User received a comment on a post.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FeedItemReceiveALike—User received a like on a post or comment.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FeedItemReceiveAnAnswer—User received an answer to a question.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FeedItemWriteAComment—User commented on a post.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FeedItemWriteAPost—User made a post.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FeedItemYourAnswerMarkedBest—User’s answer was marked as the best answer.</td>
<td></td>
</tr>
</tbody>
</table>

SEE ALSO:

- ConnectApi.UserMissionActivityCollection

ConnectApi.ActionLinkDefinition

The definition of an action link. Action link definition can be sensitive to a third party (for example, OAuth bearer token headers). For this reason, only calls made from the Apex namespace that created the action link definition can read, modify, or delete the definition. In addition, the user making the call must have created the definition or have View All Data permission.
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>actionUrl</td>
<td>String</td>
<td>The action link URL. For example, a <strong>Ui</strong> action link URL is a Web page. A <strong>Download</strong> action link URL is a link to the file to download. <strong>Ui</strong> and <strong>Download</strong> action link URLs are provided to clients. An <strong>Api</strong> or <strong>ApiAsync</strong> action link URL is a REST resource. <strong>Api</strong> and <strong>ApiAsync</strong> action link URLs aren’t provided to clients. Links to Salesforce can be relative. All other links must be absolute and start with <code>https://</code>.</td>
<td>33.0</td>
</tr>
<tr>
<td>createdDate</td>
<td>Datetime</td>
<td>ISO 8601 format date string, for example, <code>2011-02-25T18:24:31.000Z</code>.</td>
<td>33.0</td>
</tr>
<tr>
<td>excludedUserId</td>
<td>String</td>
<td>ID of a single user to exclude from performing the action. If you specify an <code>excludedUserId</code>, you can’t specify an <code>userId</code>.</td>
<td>33.0</td>
</tr>
<tr>
<td>groupDefault</td>
<td>Boolean</td>
<td><code>true</code> if this action is the default action link in the action link group; <code>false</code> otherwise. There can be only one default action link per action link group. The default action link gets distinct styling in the Salesforce UI.</td>
<td>33.0</td>
</tr>
<tr>
<td>headers</td>
<td>List&lt;ConnectApi. RequestHeader&gt;</td>
<td>The request headers for the <strong>Api</strong> and <strong>ApiAsync</strong> action link types.</td>
<td>33.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>The 18-character ID for the action link definition.</td>
<td>33.0</td>
</tr>
</tbody>
</table>
| label         | String    | A custom label to display on the action link button. A `label` value can be set only in an action link template.  

Action links have four statuses: **NewStatus**, **PendingStatus**, **SuccessStatus**, and **FailedStatus**. These strings are appended to the label for each status:  
- `label`  
- `label Pending`  
- `label Success`  
- `label Failed`  

For example, if the value of `label` is “See Example,” the values of the four action link states are: See Example, See Example Pending, See Example Success, and See Example Failed.  

An action link can use either `label` or `labelKey` to generate label names, it can’t use both. If `label` has a value, the value of `labelKey` is `None`. If `labelKey` has a value other than `None`, the value of `label` is `null`. | 34.0 |
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>labelKey</td>
<td>String</td>
<td>Key for the set of labels to show in the user interface. A set includes labels for these states: NewStatus, PendingStatus, SuccessStatus, FailedStatus. For example, if you use the Approve key, you get these labels: Approve, Pending, Approved, Failed. For a complete list of label keys, see Action Links Labels in the Connect REST API Developer Guide.</td>
<td>33.0</td>
</tr>
</tbody>
</table>
| method          | ConnectApi.HttpRequestMethod | The HTTP method. One of these values:  
  - HttpDelete—Returns HTTP 204 on success. Response body or output class is empty.  
  - HttpGet—Returns HTTP 200 on success.  
  - HttpHead—Returns HTTP 200 on success. Response body or output class is empty.  
  - HttpPatch—Returns HTTP 200 on success or HTTP 204 if the response body or output class is empty.  
  - HttpPost—Returns HTTP 201 on success or HTTP 204 if the response body or output class is empty. Exceptions are the batch posting resources and methods, which return HTTP 200 on success.  
  - HttpPut—Return HTTP 200 on success or HTTP 204 if the response body or output class is empty. | 33.0              |
| modifiedDate    | Datetime                  | ISO 8601 format date string, for example, 2011-02-25T18:24:31.000Z. | 33.0              |
| requestBody     | String                    | The request body for Api and ApiAsync action link types. **Note:** Escape quotation mark characters in the requestBody value.                                                                 | 33.0              |
| requiresConfirmation | Boolean                  | true to require the user to confirm the action; false otherwise.                                                                                                                                              | 33.0              |
| templateId      | String                    | The ID of the action link template from which to instantiate this action link. If the action link isn’t associated with a template, the value is null.                                                           | 33.0              |
| type            | ConnectApi.ActionLinkType | Defines the type of action link. Values are:  
  - Api—The action link calls a synchronous API at the action URL. Salesforce sets the status to SuccessfulStatus or FailedStatus. | 33.0              |
### ConnectApi.ActionLinkDiagnosticInfo

Any diagnostic information that may exist for an executed action link. Diagnostic info is provided only for users who can access the action link.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>diagnosticInfo</td>
<td>String</td>
<td>Any diagnostic information returned when an action link is executed. Diagnostic information is provided only for users who can access the action link.</td>
<td>33.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>The URL for this action link diagnostic information.</td>
<td>33.0</td>
</tr>
</tbody>
</table>
The definition of an action link group. Information in the action link group definition can be sensitive to a third party (for example, OAuth bearer token headers). For this reason, only calls made from the Apex namespace that created the action link group definition can read, modify, or delete the definition. In addition, the user making the call must have created the definition or have View All Data permission.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>actionLinks</td>
<td>List&lt;ConnectApi.ActionLinkDefinition&gt;</td>
<td>A collection of action link definitions that make up the action link group. Within an action link group, action links are displayed in the order listed in the actionLinks property of the ConnectApi.ActionLinkGroupDefinitionInput class. Within a feed item, action link groups are displayed in the order specified in the actionLinkGroupIds property of the ConnectApi.AssociatedActionsCapabilityInput class.</td>
<td>33.0</td>
</tr>
<tr>
<td>category</td>
<td>ConnectApi.PlatformActionGroupCategory</td>
<td>Indicates the priority and location of the action links. Values are: • Primary—The action link group is displayed in the body of the feed element. • Overflow—The action link group is displayed in the overflow menu of the feed element.</td>
<td>33.0</td>
</tr>
<tr>
<td>createdDate</td>
<td>Datetime</td>
<td>ISO 8601 date string, for example, 2011-02-25T18:24:31.000Z.</td>
<td>33.0</td>
</tr>
<tr>
<td>executions Allowed</td>
<td>ConnectApi.ActionLinkExecutionsAllowed</td>
<td>Defines the number of times an action link can be executed. Values are: • Once—An action link can be executed only one time across all users. • OncePerUser—An action link can be executed only one time for each user. • Unlimited—An action link can be executed an unlimited number of times by each user. If the action link’s actionType is Api or ApiAsync, you can’t use this value.</td>
<td>33.0</td>
</tr>
<tr>
<td>expirationDate</td>
<td>Datetime</td>
<td>ISO 8601 date string, for example, 2011-02-25T18:24:31.000Z, that represents the date and time this action group expires and can no longer be executed. If the value is null, there isn’t an expiration date.</td>
<td>33.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>18-character ID of the action link group definition.</td>
<td>33.0</td>
</tr>
<tr>
<td>modifiedDate</td>
<td>Datetime</td>
<td>ISO 8601 date string, for example, 2011-02-25T18:24:31.000Z.</td>
<td>33.0</td>
</tr>
</tbody>
</table>
ConnectApi.ActionLinkGroup

The ID of the action link group template from which to instantiate this action link group, or null if this group isn't associated with a template.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>templateId</td>
<td>String</td>
<td>The ID of the action link group template from which to instantiate this action link group, or null if this group isn't associated with a template.</td>
<td>33.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>The URL for this action link group definition.</td>
<td>33.0</td>
</tr>
</tbody>
</table>

ConnectApi.ActivitySharingResult

The results of sharing a captured email or event.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>success</td>
<td>Boolean</td>
<td>Whether the share operation succeeded or not.</td>
<td>39.0</td>
</tr>
</tbody>
</table>

ConnectApi.Actor

Actor.

This class is abstract.

Superclass of:
- ConnectApi.ActorWithId
- ConnectApi.RecommendedObject
- ConnectApi.UnauthenticatedUser

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the actor, such as the group name.</td>
<td>28.0</td>
</tr>
</tbody>
</table>
### Available Version

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>String</td>
<td>One of the following:</td>
<td>28.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• file</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• group</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• recommendedObject (version 34.0 and later)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• unauthenticateduser</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• user</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• record type name—the name of the record type, such as myCustomObject__c or Account</td>
<td></td>
</tr>
</tbody>
</table>

### See Also:
- `ConnectApi.CaseCommentCapability`
- `ConnectApi.EntityRecommendation`
- `ConnectApi.EditCapability`
- `ConnectApi.FeedEntitySummary`
- `ConnectApi.FeedItem`
- `ConnectApi.FeedItemSummary`
- `ConnectApi.Subscription`

### `ConnectApi.ActorWithId`

Actor with ID.

This class is abstract.

Subclass of `ConnectApi.Actor`.

Superclass of:
- `ConnectApi.AbstractRecordView`
- `ConnectApi.ArticleSummary`
- `ConnectApi.ChatterGroup`
- `ConnectApi.ContentHubRepository`
- `ConnectApi.File`
- `ConnectApi.RelatedFeedPost`
- `ConnectApi.User`

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>Actor’s 18-character ID</td>
<td>28.0</td>
</tr>
<tr>
<td>motif</td>
<td><code>ConnectApi.Motif</code></td>
<td>An icon that identifies the actor as a user, group, file, or custom object. The icon isn’t the user or group photo, and it isn’t a preview of the file. The motif can also contain the object’s base color.</td>
<td>28.0</td>
</tr>
</tbody>
</table>
### ConnectApi.Address

Address.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>city</td>
<td>String</td>
<td>Name of the city</td>
<td>28.0</td>
</tr>
<tr>
<td>country</td>
<td>String</td>
<td>Name of the country</td>
<td>28.0</td>
</tr>
<tr>
<td>formattedAddress</td>
<td>String</td>
<td>Formatted address per the locale of the context user</td>
<td>28.0</td>
</tr>
<tr>
<td>state</td>
<td>String</td>
<td>Name of the state, province, or so on</td>
<td>28.0</td>
</tr>
<tr>
<td>street</td>
<td>String</td>
<td>Street number</td>
<td>28.0</td>
</tr>
<tr>
<td>zip</td>
<td>String</td>
<td>Zip or postal code</td>
<td>28.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

- ConnectApi.DatacloudCompany
- ConnectApi.DatacloudContact
- ConnectApi.UserDetail

### ConnectApi.AdjustOrderSummaryOutputRepresentation

Output representation of the financial changes for an adjust items action. For a preview action, these values are the expected output. For a submit action, these values are the actual output.

Subclass of ConnectApi.BaseOutputRepresentation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>changeBalances</td>
<td>ConnectApi.ChangeItemOutputRepresentation</td>
<td>Expected (for preview) or actual (for submit) financial values for the price adjustment action. Most of the values match the change order values. If two change</td>
<td>49.0</td>
</tr>
</tbody>
</table>
orders are returned, then these values combine them. The sign of a value in this output is the opposite of the corresponding value on a change order record. For example, a discount is a positive value in changeBalances and a negative value on a change order record.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>inFulfillment ChangeOrderId</td>
<td>String</td>
<td>ID of the change Order that holds the financial changes applicable to OrderItemSummary quantities that are in the process of being fulfilled. This change Order is only created for a request that specified an allocatedItemsChangeOrderType of InFulfillment. For an adjustPreview call, this value is always null.</td>
<td>55.0</td>
</tr>
<tr>
<td>orderSummaryId</td>
<td>String</td>
<td>ID of the OrderSummary.</td>
<td>49.0</td>
</tr>
<tr>
<td>postFulfillment ChangeOrderId</td>
<td>String</td>
<td>ID of the change Order that holds the financial changes applicable to OrderItemSummary quantities that have been fulfilled. For an adjustPreview call, this value is always null.</td>
<td>49.0</td>
</tr>
<tr>
<td>preFulfillment ChangeOrderId</td>
<td>String</td>
<td>ID of the change Order that holds the financial changes applicable to OrderItemSummary quantities that have not been fulfilled. If the request specified an allocatedItemsChangeOrderType of PreFulfillment, this change Order also includes the changes applicable to OrderItemSummary quantities that are in the process of being fulfilled. For an adjustPreview call, this value is always null.</td>
<td>49.0</td>
</tr>
</tbody>
</table>

**ConnectApi.Alternative**

Alternative representation for an extension on a feed element.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>text Representation</td>
<td>String</td>
<td>Text representation of the extension.</td>
<td>40.0</td>
</tr>
<tr>
<td>thumbnailUrl</td>
<td>String</td>
<td>Thumbnail URL to the extension.</td>
<td>40.0</td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>Title of the extension.</td>
<td>40.0</td>
</tr>
</tbody>
</table>

**ConnectApi.AlternativePaymentMethodOutput**

Alternative payment method details output.
### Property Name

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>accountId</td>
<td>String</td>
<td>Salesforce Payments account to which this payment method is linked.</td>
<td>56.0</td>
</tr>
<tr>
<td>comments</td>
<td>String</td>
<td>Details about a record added by a user. Maximum of 1,000 characters.</td>
<td>56.0</td>
</tr>
<tr>
<td>email</td>
<td>String</td>
<td>Email address of the card holder.</td>
<td>56.0</td>
</tr>
<tr>
<td>gatewayToken</td>
<td>String</td>
<td>A unique, alphanumeric ID, called a token, that a payment gateway generates when it first processes a payment. The token replaces the actual payment data so that the data is kept secure. This token is stored as encrypted text, and can be used for recurring payments.</td>
<td>56.0</td>
</tr>
<tr>
<td>gatewayTokenDetails</td>
<td>String</td>
<td>Detailed information about the gateway token.</td>
<td>56.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name that you assign to the payment method object.</td>
<td>56.0</td>
</tr>
</tbody>
</table>

### ConnectApi.Announcement

An announcement displays in a designated location in the Salesforce UI until 11:59 p.m. on its expiration date, unless it's deleted or replaced by another announcement.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>expirationDate</td>
<td>Datetime</td>
<td>The Salesforce UI displays an announcement until 11:59 p.m. on this date unless another announcement is posted first. The Salesforce UI ignores the time value in the expirationDate. However, you can use the time value to create your own display logic in your own UI.</td>
<td>31.0</td>
</tr>
<tr>
<td>feedElement</td>
<td>ConnectApi.FeedElement</td>
<td>The feed element that contains the body of the announcement and its associated comments, likes, and so on.</td>
<td>31.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>18-character ID of the announcement.</td>
<td>31.0</td>
</tr>
<tr>
<td>isArchived</td>
<td>Boolean</td>
<td>Specifies whether the announcement is archived.</td>
<td>36.0</td>
</tr>
<tr>
<td>sendEmails</td>
<td>Boolean</td>
<td>Specifies whether the announcement is sent as an email to all group members.</td>
<td>36.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>The URL to the announcement.</td>
<td>33.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.AnnouncementPage
- ConnectApi.ChatterGroup
ConnectApi.AnnouncementPage

A collection of announcements.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>announcements</td>
<td>List&lt;ConnectApi.Announcement&gt;</td>
<td>A collection of ConnectApi.Announcement objects.</td>
<td>31.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>31.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or <code>null</code> if there isn’t a next page.</td>
<td>31.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the previous page, or <code>null</code> if there isn’t a previous page.</td>
<td>31.0</td>
</tr>
</tbody>
</table>

ConnectApi.ApprovalCapability

If a feed element has this capability, it includes information about an approval.

Subclass of ConnectApi.FeedElementCapability.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>The work item ID. The work item ID is <code>null</code> if there isn’t a pending work item associated with the approval record.</td>
<td>32.0</td>
</tr>
<tr>
<td>postTemplateFields</td>
<td>List&lt;ConnectApi.ApprovalPostTemplateField&gt;</td>
<td>The details of the approval post template field.</td>
<td>32.0</td>
</tr>
<tr>
<td>processInstanceStepId</td>
<td>String</td>
<td>The process instance step ID. The associated record represents one step in an approval process.</td>
<td>32.0</td>
</tr>
<tr>
<td>status</td>
<td>ConnectApi.WorkflowProcessStatus</td>
<td>The status of the approval.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.FeedElementCapabilities

ConnectApi.ApprovalIntent

Approval intent for a social post.
### isRecallable

- **Type**: Boolean
- **Description**: Specifies whether the social post can be recalled (`true`) or not (`false`).
- **Available Version**: 45.0

**SEE ALSO:**
- `ConnectApi.SocialPostIntents`

### ConnectApi.ApprovalPostTemplateField

Approval post template field.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>displayName</td>
<td>String</td>
<td>The field name.</td>
<td>28.0</td>
</tr>
<tr>
<td>displayValue</td>
<td>String</td>
<td>The field value or <code>null</code> if the field is set to <code>null</code>.</td>
<td>28.0</td>
</tr>
<tr>
<td>record</td>
<td>ConnectApi.Reference</td>
<td>A record ID. If no record exists or if the reference is <code>null</code>, this value is <code>null</code>.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- `ConnectApi.ApprovalCapability`

### ConnectApi.ArticleItem

Article item in question and answers suggestions.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>Id of the article.</td>
<td>32.0</td>
</tr>
<tr>
<td>rating</td>
<td>Double</td>
<td>The rating of the article.</td>
<td>32.0</td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>Title of the article.</td>
<td>32.0</td>
</tr>
<tr>
<td>urlLink</td>
<td>String</td>
<td>Link URL of the article.</td>
<td>32.0</td>
</tr>
<tr>
<td>viewCount</td>
<td>Integer</td>
<td>Number of votes given to the article.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- `ConnectApi.QuestionAndAnswersSuggestions`

### ConnectApi.ArticleSummary

A knowledge article summary.

Subclass of `ConnectApi.ActorWithId`. 
ConnectApi.AssociatedActionsCapability

If a feed element has this capability, it has platform actions associated with it.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>platformActionGroups</td>
<td>List&lt;ConnectApi.PlatformActionGroup&gt;</td>
<td>The platform action groups associated with a feed element. Platform action groups are returned in the order specified in the ConnectApi.AssociatedActionsCapabilityInput class.</td>
<td>33.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

ConnectApi.FeedElementCapabilities

ConnectApi.AsyncOutputRepresentation

Output representation of the async operation.

Subclass of ConnectApi.BaseAsyncOutputRepresentation.

No additional properties.

SEE ALSO:

multipleEnsureFundsAsync(multipleEnsureFundsInput)
ConnectApi.MultipleAsyncOutputRepresentation

ConnectApi.Audience

A personalization audience.
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>criteria</td>
<td>List&lt;ConnectApi.AudienceCriteriaDetail&gt;</td>
<td>Criteria details for the audience.</td>
<td>48.0</td>
</tr>
<tr>
<td>customFormula</td>
<td>String</td>
<td>Custom formula for the audience criteria. For example, (1 AND 2) OR 3.</td>
<td>48.0</td>
</tr>
<tr>
<td>formulaFilterType</td>
<td>ConnectApi.FormulaFilterType</td>
<td>Formula filter type for the personalization audience. Values are:</td>
<td>48.0</td>
</tr>
<tr>
<td></td>
<td>• AllCriteriaMatch—All audience criteria are true (AND operation).</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• AnyCriterionMatches—Any audience criterion is true (OR operation).</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• CustomLogicMatches—Audience criteria match the custom formula (for example, (1 AND 2) OR 3).</td>
<td></td>
<td></td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the audience.</td>
<td>48.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the audience.</td>
<td>48.0</td>
</tr>
<tr>
<td>targets</td>
<td>List&lt;ConnectApi.AudienceTargetAssignment&gt;</td>
<td>Target assignments for the audience.</td>
<td>48.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>URL to this audience.</td>
<td>48.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.AudienceCollection

### ConnectApi.AudienceCollection
Collection of personalization audiences.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>audiences</td>
<td>List&lt;ConnectApi.Audience&gt;</td>
<td>Collection of audiences.</td>
<td>48.0</td>
</tr>
</tbody>
</table>

### ConnectApi.AudienceCriteria
Custom recommendation audience criteria.
This class is abstract.
This class is a superclass of:
- ConnectApi.CustomListAudienceCriteria
- ConnectApi.NewUserAudienceCriteria
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
</table>
| type          | ConnectApi.RecommendationAudienceCriteriaType | Specifies the custom recommendation audience criteria type. One of these values:  
• CustomList—A custom list of users makes up the audience.  
• MaxDaysInCommunity—New members make up the audience. | 36.0 |

**SEE ALSO:**  
ConnectApi.RecommendationAudience

### ConnectApi.AudienceCriteriaDetail

Personalization audience criteria.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>criterion</td>
<td>List&lt;ConnectApi.AudienceCriterionDetail&gt;</td>
<td>List of mappings of audience criteria fields and values.</td>
<td>48.0</td>
</tr>
<tr>
<td>criterionNumber</td>
<td>Integer</td>
<td>Number associated with the audience criterion in a formula. For example, (1 AND 2) OR 3. If unspecified, criteria are assigned numbers in the order that they're added.</td>
<td>48.0</td>
</tr>
</tbody>
</table>
| criterionOperator | ConnectApi.AudienceCriteriaOperator | Operator used in the personalization audience criterion. Values are:  
• Contains  
• Equal  
• GreaterThan  
• GreaterThanOrEqual  
• Includes  
• LessThan  
• LessThanOrEqual  
• NotEqual  
• NotIncludes  
• StartsWith | 48.0 |
**ConnectApi.AudienceCriterionDetail**

Audience criterion information.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>value</td>
<td>Map&lt;String, String&gt;</td>
<td>Mapping of an audience criterion value and field.</td>
<td>48.0</td>
</tr>
</tbody>
</table>

**ConnectApi.AudienceTarget**

Personalization audience assigned to a target.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>audienceName</td>
<td>String</td>
<td>Name of the audience assigned to the target.</td>
<td>48.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the audience assigned to the target.</td>
<td>48.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>URL to the audience assigned to the target.</td>
<td>48.0</td>
</tr>
</tbody>
</table>
## ConnectApi.AudienceTargetAssignment

Target assignments for a personalization audience.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>formulaScope</td>
<td>ConnectApi.FormulaScope</td>
<td>Formula scope of the target.</td>
<td>51.0</td>
</tr>
<tr>
<td>groupName</td>
<td>String</td>
<td>Group name of the target. Groups bundle related target and audience pairs.</td>
<td>48.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the target.</td>
<td>48.0</td>
</tr>
<tr>
<td>isMatch</td>
<td>Boolean</td>
<td>Specifies whether the target matches the current context (true) or doesn't (false).</td>
<td>48.0</td>
</tr>
<tr>
<td>priority</td>
<td>Integer</td>
<td>Priority of the target. Within a group, priority determines which target is returned if the user matches more than one audience.</td>
<td>48.0</td>
</tr>
<tr>
<td>publishStatus</td>
<td>ConnectApi.PublishStatus</td>
<td>Publish status of the target. Values are:</td>
<td>48.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Draft</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Live</td>
<td></td>
</tr>
<tr>
<td>targetType</td>
<td>String</td>
<td>Type of target, indicating the nature of the data being targeted.</td>
<td>48.0</td>
</tr>
<tr>
<td>targetValue</td>
<td>String</td>
<td>Value of the target.</td>
<td>48.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>URL to the target.</td>
<td>48.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.Audience

## ConnectApi.AvailableLocationOutputRepresentation

A set of inventory locations that can combine to fulfill an order.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>locations</td>
<td>List&lt;String&gt;</td>
<td>A list of inventory locations.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- findRoutesWithFewestSplits(findRoutesWithFewestSplitsInputRepresentation)
- findRoutesWithFewestSplitsUsingOCI(findRoutesWithFewestSplitsUsingOCIInput)
- ConnectApi.FindRoutesWithFewestSplitsOutputRepresentation
- ConnectApi.FindRoutesWithFewestSplitsUsingOCIOutputRepresentation
**ConnectApi.AverageDistanceResultOutputRepresentation**

Wraps inventory location shipping distance calculation results.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>distanceCalculation</td>
<td>ConnectApi.DistanceCalculationOutputRepresentation</td>
<td>Results of the shipping distance calculations.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

**ConnectApi.PaymentAuthAdjustmentResponse**


<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>accountId</td>
<td>String</td>
<td>ID of the account containing the payment authorization being adjusted.</td>
<td>51.0</td>
</tr>
<tr>
<td>amount</td>
<td>Double</td>
<td>Amount of the payment authorization adjustment.</td>
<td>51.0</td>
</tr>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Three-letter ISO 4217 currency code associated with the payment authorization adjustment.</td>
<td>51.0</td>
</tr>
<tr>
<td>effectiveDate</td>
<td>Datetime</td>
<td>Date when the authorization adjustment becomes effective.</td>
<td>51.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the PaymentAuthAdjustment record.</td>
<td>51.0</td>
</tr>
<tr>
<td>paymentAuthAdjustNumber</td>
<td>String</td>
<td>System-defined reference number.</td>
<td>51.0</td>
</tr>
<tr>
<td>requestDate</td>
<td>Datetime</td>
<td>Date when the authorization adjustment transaction occurred.</td>
<td>51.0</td>
</tr>
<tr>
<td>status</td>
<td>String</td>
<td>Status of the payment authorization adjustment. Possible values are:</td>
<td>51.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Canceled: The payment authorization reversal has been canceled. The parent authorization has returned to its pre-reversal balance.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Draft: The payment authorization reversal can be edited before applying it against the parent authorization.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Processed: The payment authorization reversal has been finalized.</td>
<td></td>
</tr>
</tbody>
</table>

Users can change the status as follows:

- Draft to Processed
- Processed to Canceled
- Draft to Canceled
ConnectApi.AuthReversalGatewayResponse

Authorization Reversal Gateway Response Representation.
No additional properties.

ConnectApi.AuthorizationReversalResponse

Authorization Reversal output representation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>error</td>
<td><code>ConnectApi ErrorResponse</code></td>
<td>Error response representation for the authorization reversal.</td>
<td>51.0</td>
</tr>
<tr>
<td>gatewayResponse</td>
<td><code>ConnectApi AuthReversalGatewayResponse</code></td>
<td>Gateway response representation for authorization reversal.</td>
<td>51.0</td>
</tr>
<tr>
<td>paymentAuthAdjustment</td>
<td><code>ConnectApi PaymentAuthAdjustmentResponse</code></td>
<td>Payment authorization adjustment response representation for the authorization reversal.</td>
<td>51.0</td>
</tr>
<tr>
<td>paymentGatewayLogs</td>
<td><code>List&lt;ConnectApi GatewayLogResponse&gt;</code></td>
<td>Gateway log collection representation for the authorization reversal.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

ConnectApi.AuthorizationGatewayResponse

Payment gateway authorization response representation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>gatewayAuthorizationCode</td>
<td><code>String</code></td>
<td>Gateway authorization code.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

ConnectApi.AuthorizationResponse

Payment Authorization output representation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>error</td>
<td><code>ConnectApi ErrorResponse</code></td>
<td>Error representation for the payment authorization.</td>
<td>51.0</td>
</tr>
<tr>
<td>gatewayResponse</td>
<td><code>ConnectApi AuthorizationGatewayResponse</code></td>
<td>Gateway response representation for the payment authorization.</td>
<td>51.0</td>
</tr>
</tbody>
</table>
### ConnectApi.BannerCapability

If a feed element has this capability, it has a banner motif and style.

Subclass of `ConnectApi.FeedElementCapability`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>motif</td>
<td><code>ConnectApi.Motif</code></td>
<td>A banner motif.</td>
<td>31.0</td>
</tr>
</tbody>
</table>
| style         | `ConnectApi.BannerStyle` | Decorates a feed item with a color and set of icons. Possible value:  
- Announcement—An announcement displays in a designated location in the Salesforce UI until 11:59 p.m. on its expiration date, unless it's deleted or replaced by another announcement. | 31.0 |

**SEE ALSO:**  
`ConnectApi.FeedElementCapabilities`

### ConnectApi.BannerPhoto

A banner photo.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>bannerPhotoUrl</code></td>
<td><code>String</code></td>
<td>URL to the banner photo in a large format. This URL is available only to authenticated users.</td>
<td>36.0</td>
</tr>
<tr>
<td><code>bannerPhotoVersionId</code></td>
<td><code>String</code></td>
<td>18-character version ID of the banner photo.</td>
<td>36.0</td>
</tr>
</tbody>
</table>
### ConnectApi.BaseManagedSocialAccount

Base information describing a managed social account or fan page of a social network.

This class is abstract.

Superclass of `ConnectApi.ManagedSocialAccount`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>defaultResponse</td>
<td>String</td>
<td>Default response account to use when replying to posts sent to this account.</td>
<td>44.0</td>
</tr>
<tr>
<td>AccountId</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>displayName</td>
<td>String</td>
<td>Real name (or user name if real name not available) for this account on the social network.</td>
<td>44.0</td>
</tr>
<tr>
<td>externalPictureUrl</td>
<td>String</td>
<td>URL to the account’s avatar image.</td>
<td>44.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>Internal SFDC ID for this managed social account.</td>
<td>44.0</td>
</tr>
<tr>
<td>label</td>
<td>String</td>
<td>Label for the social account.</td>
<td>44.0</td>
</tr>
<tr>
<td>profileUrl</td>
<td>String</td>
<td>URL to the account’s profile.</td>
<td>44.0</td>
</tr>
<tr>
<td>socialNetwork</td>
<td>ConnectApi.</td>
<td>Social network that this account belongs to. Values are:</td>
<td>44.0</td>
</tr>
<tr>
<td></td>
<td>SocialNetworkProvider</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Facebook</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- GooglePlus</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Instagram</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- InstagramBusiness</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- KakaoTalk</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Kik</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Line</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- LinkedIn</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Messenger</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Other</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Pinterest</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- QQ</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Rypple</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- SinaWeibo</td>
<td></td>
</tr>
</tbody>
</table>

SEE ALSO:
- `ConnectApi.ChatterGroup`
- `ConnectApi.UserDetail`
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>• SMS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Snapchat</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Telegram</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Twitter</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• VKontakte</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• WeChat</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• WhatsApp</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• YouTube</td>
<td></td>
</tr>
<tr>
<td>uniqueName</td>
<td>String</td>
<td>Unique name used for distinguishing same name fan pages; acts like a user name for a fan page.</td>
<td>44.0</td>
</tr>
<tr>
<td>username</td>
<td>String</td>
<td>Unique user name or handle for this account on the social network.</td>
<td>44.0</td>
</tr>
</tbody>
</table>

**ConnectApi.BaseAsyncOutputRepresentation**

Base Order Management async output class.

This class is abstract.

Subclass of `ConnectApi.BaseOutputRepresentation`.

Superclass of:
- `ConnectApi.AsyncOutputRepresentation`
- `ConnectApi.EnsureFundsAsyncOutputRepresentation`
- `ConnectApi.EnsureRefundsAsyncOutputRepresentation`

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>backgroundOperationId</td>
<td>String</td>
<td>ID of the background operation.</td>
<td>48.0</td>
</tr>
</tbody>
</table>

**ConnectApi.BaseInvoiceOutputRepresentation**

Base Order Management Invoice output class.

This class is abstract.

Subclass of `ConnectApi.BaseOutputRepresentation`.

Superclass of `ConnectApi.ChangeOrdersInvoiceOutputRepresentation`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>invoiceId</td>
<td>String</td>
<td>ID of the created invoice.</td>
<td>56.0</td>
</tr>
</tbody>
</table>
ConnectApi.BaseOutputRepresentation

Base Order Management output class.

This class is abstract.

Superclass of:

- ConnectApi.AdjustOrderSummaryOutputRepresentation
- ConnectApi.BaseAsyncOutputRepresentation
- ConnectApi.BaseInvoiceOutputRepresentation
- ConnectApi.ConfirmHeldFOCapacityOutputRepresentation
- ConnectApi.CreateCreditMemoOutputRepresentation
- ConnectApi.CreateMultipleInvoicesFromChangeOrdersOutputRepresentation
- ConnectApi.CreateOrderPaymentSummaryOutputRepresentation
- ConnectApi.EnsureFundsAsyncOutputRepresentation
- ConnectApi.EnsureRefundsAsyncOutputRepresentation
- ConnectApi.FindRoutesWithFewestSplitsOutputRepresentation
- ConnectApi.FindRoutesWithFewestSplitsUsingOCIOutputRepresentation
- ConnectApi.FindRoutesWithFewestSplitsWithInventoryOutputRepresentation
- ConnectApi.FulfillmentGroupOutputRepresentation
- ConnectApi.FulfillmentOrderCancelLineItemsOutputRepresentation
- ConnectApi.FulfillmentOrderInvoiceOutputRepresentation
- ConnectApi.FulfillmentOrderOutputRepresentation
- ConnectApi.GetFOCapacityValuesOutputRepresentation
- ConnectApi.HoldFOCapacityOutputRepresentation
- ConnectApi.MultipleAsyncOutputRepresentation
- ConnectApi.MultipleFulfillmentOrderInvoicesOutputRepresentation
- ConnectApi.MultipleFulfillmentOrderOutputRepresentation
- ConnectApi.OrderSummaryOutputRepresentation
- ConnectApi.PreviewCancelOutputRepresentation
- ConnectApi.PreviewReturnOutputRepresentation
- ConnectApi.ProductDetailsOutputRepresentation
- ConnectApi.RankAverageDistanceOutputRepresentation
- ConnectApi.ReleaseHeldFOCapacityOutputRepresentation
- ConnectApi.ReturnItemsOutputRepresentation
- ConnectApi.ReturnOrderItemSplitLineOutputRepresentation
- ConnectApi.ReturnOrderOutputRepresentation
- ConnectApi.SubmitCancelOutputRepresentation
- ConnectApi.SubmitReturnOutputRepresentation
### Available Version

**Property Name** | **Type** | **Description** | **Available Version**
--- | --- | --- | ---
errors &nbsp; | List<ConnectApi.ErrorResponse> &nbsp; | Any errors that were returned. | 48.0 |
success &nbsp; | Boolean &nbsp; | Indicates whether the transaction was successful. | 48.0 |

---

### ConnectApi.BatchResult

The result of an operation returned by a batch method.

#### Usage

Calls to batch methods return a list of BatchResult objects. Each element in the BatchResult list corresponds to the strings in the list parameter passed to the batch method. The first element in the BatchResult list matches the first string passed in the list parameter, the second element corresponds with the second string, and so on. If only one string is passed, the BatchResult list contains a single element.

#### Example

The following example shows how to obtain and iterate through the returned ConnectApi.BatchResult objects. The code adds two group IDs to a list. One of group IDs is incorrect, which causes a failure when the code calls the batch method. After it calls the batch method, it iterates through the results to determine whether the operation was successful or not for each group ID in the list. The code writes the ID of every group that was processed successfully to the debug log. The code writes an error message for every failed group.

This example generates one successful operation and one failure.

```java
List<String> myList = new List<String>(){
    // Add one correct group ID.
    myList.add('0F9D00000000oOT');
    // Add one incorrect group ID.
    myList.add('0F9D00000000izf');

    ConnectApi.BatchResult[] batchResults = ConnectApi.ChatterGroups.getGroupBatch(null, myList);

    // Iterate through each returned result.
    for (ConnectApi.BatchResult batchResult : batchResults) {
        if (batchResult.isSuccess()) {
            // Operation was successful.
            // Print the group ID.
            ConnectApi.ChatterGroupSummary groupSummary;
            if(batchResult.getResult() instanceof ConnectApi.ChatterGroupSummary) {
                groupSummary = (ConnectApi.ChatterGroupSummary) batchResult.getResult();
            }
            System.debug('SUCCESS');
            System.debug(groupSummary.id);
        } else {
            // Operation failed. Print errors.
            System.debug('FAILURE');
        }
    }
};
```
IN THIS SECTION:

**BatchResult Methods**

These are instance methods for **BatchResult**.

**getError()**

If an error occurred, returns a **ConnectApi.ConnectApiException** object providing the error code and description.

**getErrorMessage()**

Returns a String that contains an error message.

**getErrorTypeName()**

Returns a String that contains the name of the error type.

**getResult()**

Returns an object that contains the results of the batch operation. The object is typed according to the batch method. For example, if you call `getMembershipBatch()`, a successful call to BatchResult `getResult()` returns a **ConnectApi.GroupMembership** object.

**isSuccess()**

Returns a Boolean that is set to `true` if the batch operation was successful for this object, `false` otherwise.

**getError()**

If an error occurred, returns a **ConnectApi.ConnectApiException** object providing the error code and description.

**getErrorMessage()**

Returns a String that contains an error message.

**getErrorTypeName()**

Returns a String that contains the name of the error type.

**getResult()**

Returns an object that contains the results of the batch operation. The object is typed according to the batch method. For example, if you call `getMembershipBatch()`, a successful call to BatchResult `getResult()` returns a **ConnectApi.GroupMembership** object.

**isSuccess()**

Returns a Boolean that is set to `true` if the batch operation was successful for this object, `false` otherwise.
Usage
The error message doesn't make a round trip through a Visualforce view state, because exceptions can't be serialized.

getErrorTypeName()
Returns a String that contains the name of the error type.

Signature

```java
public String getErrorTypeName()
```

ReturnValue
Type: String

getResult()
Returns an object that contains the results of the batch operation. The object is typed according to the batch method. For example, if you call `getMembershipBatch()`, a successful call to `BatchResult getResult()` returns a `ConnectApi.GroupMembership` object.

Signature

```java
public Object getResult()
```

ReturnValue
Type: Object

isSuccess()
Returns a Boolean that is set to `true` if the batch operation was successful for this object, `false` otherwise.

Signature

```java
public Boolean isSuccess()
```

ReturnValue
Type: Boolean

ConnectApi.BlankRecordField
Record field displayed as a place holder in a grid of fields.
Subclass of `ConnectApi.AbstractRecordField`. 
**ConnectApi.BookmarksCapability**

If a feed element has this capability, the context user can bookmark it.
Subclass of `ConnectApi.FeedElementCapability`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isBookmarkedByCurrentUser</td>
<td>Boolean</td>
<td>Indicates whether the feed element has been bookmarked by the context user (true) or not (false).</td>
<td>32.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- `ConnectApi.FeedElementCapabilities`

**ConnectApi.BookmarkSummary**

Summary of a bookmark.
Subclass of `ConnectApi.UserFeedEntityActivitySummary`.
No additional properties.

**ConnectApi.BotInfoRepresentation**

Information about the bot associated with the conversation application.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>botId</td>
<td>String</td>
<td>ID of the bot.</td>
<td>54.0</td>
</tr>
<tr>
<td>botName</td>
<td>String</td>
<td>Name of the bot.</td>
<td>54.0</td>
</tr>
<tr>
<td>lastModifiedDate</td>
<td>Datetime</td>
<td>Last modified date of the bot definition.</td>
<td>54.0</td>
</tr>
</tbody>
</table>

**ConnectApi.BotVersionActivationInfo**

Success or failure information of the bot version activation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isActivated</td>
<td>Boolean</td>
<td>Indicates whether the bot is active or not.</td>
<td>51.0</td>
</tr>
<tr>
<td>messages</td>
<td>List&lt;String&gt;</td>
<td>Failure messages.</td>
<td>50.0</td>
</tr>
<tr>
<td>success</td>
<td>Boolean</td>
<td>Indicates whether the activation was successful or not.</td>
<td>50.0</td>
</tr>
</tbody>
</table>

**ConnectApi.BundleCapability**

If a feed element has this capability, it has a container of feed elements called a **bundle**.
This class is abstract.

Subclass of `ConnectApi.FeedElementCapability`.

Superclass of:
- `ConnectApi.GenericBundleCapability`
- `ConnectApi.TrackedChangeBundleCapability`

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>bundleType</td>
<td><code>ConnectApi.BundleType</code></td>
<td>Defines this feed element’s bundle type. The bundle type determines what additional information appears in the bundle.</td>
<td>31.0</td>
</tr>
<tr>
<td>page</td>
<td><code>ConnectApi.FeedElementPage</code></td>
<td>A collection of feed elements.</td>
<td>31.0</td>
</tr>
<tr>
<td>totalElements</td>
<td><code>Integer</code></td>
<td>The total number of feed elements that this bundle aggregates.</td>
<td>31.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- `ConnectApi.FeedElementCapabilities`

### ConnectApi.CallCollaborationCapability

If a feed element has this capability, it has a recording comment.

Subclass of `ConnectApi.FeedElementCapability`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>commentEndTime</td>
<td><code>Integer</code></td>
<td>End time of the comment on the media player, in seconds.</td>
<td>51.0</td>
</tr>
<tr>
<td>commentStartTime</td>
<td><code>Integer</code></td>
<td>Start time of the comment on the media player, in seconds.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- `ConnectApi.FeedElementCapabilities`

### ConnectApi.CandidateAnswersStatus

The status of candidate answers on a feed element.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>hasCandidateAnswers</td>
<td><code>Boolean</code></td>
<td>Indicates whether candidate answers are available for a question.</td>
<td>41.0</td>
</tr>
<tr>
<td>hasCandidateAnswersPublished</td>
<td><code>Boolean</code></td>
<td>Indicates whether any candidate answers are published.</td>
<td>41.0</td>
</tr>
</tbody>
</table>
### ConnectApi.CanvasCapability

If a feed element has this capability, it renders a canvas app.

Subclass of `ConnectApi.FeedElementCapability`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>String</td>
<td>A description of the canvas app. The maximum size is 255 characters.</td>
<td>32.0</td>
</tr>
<tr>
<td>developerName</td>
<td>String</td>
<td>The API name (developer name) of the connected app.</td>
<td>32.0</td>
</tr>
<tr>
<td>height</td>
<td>String</td>
<td>The height of the canvas app in pixels.</td>
<td>32.0</td>
</tr>
<tr>
<td>icon</td>
<td>ConnectApi.Icon</td>
<td>The icon for the canvas app.</td>
<td>32.0</td>
</tr>
<tr>
<td>namespacePrefix</td>
<td>String</td>
<td>A unique namespace prefix for the canvas app.</td>
<td>32.0</td>
</tr>
<tr>
<td>parameters</td>
<td>String</td>
<td>JSON parameters passed to the canvas app.</td>
<td>32.0</td>
</tr>
<tr>
<td>thumbnailUrl</td>
<td>String</td>
<td>A thumbnail URL to a preview image. The maximum thumbnail size is 120 pixels</td>
<td>32.0</td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>A title for the canvas link.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

- `ConnectApi.QuestionAndAnswersCapability`
- `ConnectApi.FeedElementCapabilities`

### ConnectApi.CapacityResponseOutputRepresentation

Response to a request related to a location’s fulfillment capacity.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>actionRequestId</td>
<td>String</td>
<td>Unique string that identifies the original capacity request.</td>
<td>55.0</td>
</tr>
<tr>
<td>error</td>
<td>ConnectApi.ErrorResponse</td>
<td>Error returned by the request, if any.</td>
<td>55.0</td>
</tr>
<tr>
<td>success</td>
<td>Boolean</td>
<td>Indicates whether the request was successful.</td>
<td>55.0</td>
</tr>
</tbody>
</table>
ConnectApi.CaptureGatewayResponse

Gateway capture response.
No additional properties.

ConnectApi.CaptureResponse

Capture output.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>error</td>
<td>ConnectApi.ErrorResponse</td>
<td>Error response representation for an authorization capture.</td>
<td>50.0</td>
</tr>
<tr>
<td>gatewayResponse</td>
<td>ConnectApi.CaptureGatewayResponse</td>
<td>Gateway log response containing details about gateway logs created during the process of the capture request.</td>
<td>50.0</td>
</tr>
<tr>
<td>payment</td>
<td>ConnectApi.PaymentResponse</td>
<td>Payment response object for the capture request. Contains the information related to a payment object created during request processing.</td>
<td>50.0</td>
</tr>
<tr>
<td>paymentGatewayLogs</td>
<td>List&lt;ConnectApi.GatewayLogResponse&gt;</td>
<td>Gateway log collection for an authorization capture.</td>
<td>50.0</td>
</tr>
<tr>
<td>paymentGroup</td>
<td>ConnectApi.PaymentGroupResponse</td>
<td>Payment group associated with the capture request.</td>
<td>50.0</td>
</tr>
</tbody>
</table>

ConnectApi.CardPaymentMethodOutput

Card payment method details output.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>accountId</td>
<td>String</td>
<td>Salesforce Payments account to which this payment method is linked.</td>
<td>56.0</td>
</tr>
<tr>
<td>autoPay</td>
<td>Boolean</td>
<td>Indicates whether a token for recurring payments is being requested (true) or not (false). The token lets the payment method be used for recurring payments.</td>
<td>56.0</td>
</tr>
<tr>
<td>cardBin</td>
<td>String</td>
<td>Bank Identification Number (BIN). The BIN is the first 4-6 numbers on a payment card that identifies the card issuer.</td>
<td>56.0</td>
</tr>
<tr>
<td>cardCategory</td>
<td>ConnectApi.CardCategory</td>
<td>• CreditCard • DebitCard</td>
<td>56.0</td>
</tr>
<tr>
<td>cardHolderFirstName</td>
<td>String</td>
<td>First name of the card holder</td>
<td>56.0</td>
</tr>
</tbody>
</table>
### Available Version

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>cardHolderLastName</td>
<td>String</td>
<td>Last name of the card holder</td>
<td>56.0</td>
</tr>
<tr>
<td>cardHolderName</td>
<td>String</td>
<td>Full name of card holder</td>
<td>56.0</td>
</tr>
<tr>
<td>cardLastFour</td>
<td>String</td>
<td>Last four digits on a card</td>
<td>56.0</td>
</tr>
<tr>
<td>cardType</td>
<td>ConnectApi.CardType</td>
<td>Credit card issuer.</td>
<td>56.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AmericanExpress</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• DinersClub</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• JCB</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Maestro</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• MasterCard</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Visa</td>
<td></td>
</tr>
<tr>
<td>comments</td>
<td>String</td>
<td>Details about a record added by a user. Maximum of 1,000 characters.</td>
<td>56.0</td>
</tr>
<tr>
<td>displayCardNumber</td>
<td>String</td>
<td>Card displayed number</td>
<td>56.0</td>
</tr>
<tr>
<td>email</td>
<td>String</td>
<td>Email address of the card holder</td>
<td>56.0</td>
</tr>
<tr>
<td>expiryMonth</td>
<td>Integer</td>
<td>Card expiration month</td>
<td>56.0</td>
</tr>
<tr>
<td>expiryYear</td>
<td>Integer</td>
<td>Card expiration year</td>
<td>56.0</td>
</tr>
<tr>
<td>nickName</td>
<td>String</td>
<td>Optional card nickname</td>
<td>56.0</td>
</tr>
<tr>
<td>startMonth</td>
<td>Integer</td>
<td>Month when card becomes active</td>
<td>56.0</td>
</tr>
<tr>
<td>startYear</td>
<td>Integer</td>
<td>Year when card becomes active</td>
<td>56.0</td>
</tr>
</tbody>
</table>

#### ConnectApi.CartCoupon

Cart Coupon representation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>cartCouponId</td>
<td>String</td>
<td>ID of the cart coupon code.</td>
<td>54.0</td>
</tr>
<tr>
<td>couponCode</td>
<td>String</td>
<td>The coupon code a buyer can use to manually apply a promotion to the cart.</td>
<td>54.0</td>
</tr>
</tbody>
</table>

#### ConnectApi.CartCouponCollection

Collection of coupons related to a cart.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>cartCoupons</td>
<td>ConnectApi.CartCouponList&gt;</td>
<td>Collection of coupons.</td>
<td>54.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>-------------------</td>
<td>--------------------------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>cartId</td>
<td>String</td>
<td>ID of the cart.</td>
<td>54.0</td>
</tr>
<tr>
<td>cartStatus</td>
<td>ConnectApi.CartStatus</td>
<td>Status of the cart. Values are:</td>
<td>54.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Active—Cart is created and available for modifications, like adding or removing products or promotions.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Checkout—Cart is in checkout. If the customer modifies the cart, the current checkout session is canceled.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Closed—Checkout is complete and an order was created. The cart cannot be modified.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- PendingClosed—Cart is marked to be closed, but the request isn’t completed yet. The cart can’t be modified. This value is available in API version 57.0 and later.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- PendingDelete—Cart is marked for delete, but the request isn’t completed yet. The cart can’t be modified.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Processing—Cart is processing. For example, taxes are being calculated. The cart can’t be modified.</td>
<td></td>
</tr>
<tr>
<td>ownerId</td>
<td>String</td>
<td>ID of the user who owns the cart.</td>
<td>54.0</td>
</tr>
</tbody>
</table>

### ConnectApi.CartCouponList

List of coupons for a cart.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>coupons</td>
<td>List&lt;ConnectApi.CartCoupon&gt;</td>
<td>List of coupons associated with a cart.</td>
</tr>
</tbody>
</table>

### ConnectApi.CartInventoryItemReservationOutputRepresentation (Pilot)

Inventory item reservation.

**Note:** This feature is not generally available and is being piloted with certain Customers subject to additional terms and conditions. It is not part of your purchased Services. This feature is subject to change, may be discontinued with no notice at any time in Salesforce’s sole discretion, and Salesforce may never make this feature generally available. Make your purchase decisions only on the basis of generally available products and features. This feature is made available on an AS IS basis and use of this feature is at your sole risk.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>errorCode</td>
<td>String</td>
<td>Error code for this reservation item.</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>---------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>errorMessage</td>
<td>String</td>
<td>Error message for this reservation item.</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID for this reservation item.</td>
</tr>
<tr>
<td>itemReservation</td>
<td>String</td>
<td>Item reservation source ID for this reservation item.</td>
</tr>
<tr>
<td>LocationId</td>
<td>String</td>
<td>Reserved at location or group ID for this reservation item.</td>
</tr>
<tr>
<td>productId</td>
<td>String</td>
<td>Product ID for this reservation item.</td>
</tr>
<tr>
<td>quantity</td>
<td>Double</td>
<td>Quantity for this reservation item.</td>
</tr>
</tbody>
</table>

**ConnectApi.CartInventoryReservationOutputRepresentation (Pilot)**

Inventory Reservation

**Note:** This feature is not generally available and is being piloted with certain Customers subject to additional terms and conditions. It is not part of your purchased Services. This feature is subject to change, may be discontinued with no notice at any time in Salesforce’s sole discretion, and Salesforce may never make this feature generally available. Make your purchase decisions only on the basis of generally available products and features. This feature is made available on an AS IS basis and use of this feature is at your sole risk.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>errorCode</td>
<td>String</td>
<td>Error code for this reservation.</td>
<td>58.0</td>
</tr>
<tr>
<td>errorMessage</td>
<td>String</td>
<td>Error message for this reservation.</td>
<td>58.0</td>
</tr>
<tr>
<td>inventoryItem</td>
<td>List&lt;ConnectApi.CartInventory</td>
<td>Collection of inventory item reservations.</td>
<td>58.0</td>
</tr>
<tr>
<td></td>
<td>ItemReservationOutputRepresentation&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>reservationIdentifier</td>
<td>String</td>
<td>Reservation identifier.</td>
<td>58.0</td>
</tr>
<tr>
<td>success</td>
<td>Boolean</td>
<td>Indicates whether the transaction was successful.</td>
<td>58.0</td>
</tr>
</tbody>
</table>

**ConnectApi.CartItem**

An item in a cart.
Subclass of ConnectApi.AbstractCartItem.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Currency ISO code of the cart.</td>
<td>57.0</td>
</tr>
<tr>
<td>itemizedAdjustment</td>
<td>String</td>
<td>Total itemized adjustment amount for the item, including promotions and excluding taxes.</td>
<td>52.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>---------------------</td>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>listPrice</td>
<td>String</td>
<td>List price for the item.</td>
<td>49.0</td>
</tr>
<tr>
<td>salesPrice</td>
<td>String</td>
<td>Sales price for the item.</td>
<td>49.0</td>
</tr>
<tr>
<td>totalAdjustment</td>
<td>String</td>
<td>Total adjustment amount for the item.</td>
<td>50.0</td>
</tr>
<tr>
<td>totalAmount</td>
<td>String</td>
<td>Total amount for the item.</td>
<td>49.0</td>
</tr>
<tr>
<td>totalListPrice</td>
<td>String</td>
<td>Total list price for the item.</td>
<td>49.0</td>
</tr>
<tr>
<td>totalPrice</td>
<td>String</td>
<td>Total price for the item including adjustments but excluding taxes.</td>
<td>49.0</td>
</tr>
<tr>
<td>totalTax</td>
<td>String</td>
<td>Total tax for the item.</td>
<td>49.0</td>
</tr>
<tr>
<td>unitAdjustedPrice</td>
<td>String</td>
<td>Unit price, including adjustments, for the item. This value is informational only and isn’t used in pricing calculations.</td>
<td>50.0</td>
</tr>
<tr>
<td>unitAdjustment</td>
<td>String</td>
<td>Adjustments made to the unit price for the item. This value is informational only and isn’t used in pricing calculations.</td>
<td>50.0</td>
</tr>
</tbody>
</table>

**ConnectApi.CartItemCollection**

Cart item collection.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>cartCoupons</td>
<td>ConnectApi.CartCouponCollection</td>
<td>Collection of coupons in the cart.</td>
<td>59.0</td>
</tr>
<tr>
<td>cartItems</td>
<td>List&lt;ConnectApi.CartItemResult&gt;</td>
<td>Collection of cart item results.</td>
<td>49.0</td>
</tr>
<tr>
<td>cartPromotions</td>
<td>ConnectApi.CartPromotionCollection</td>
<td>Collection of promotions in the cart.</td>
<td>59.0</td>
</tr>
<tr>
<td>cartSummary</td>
<td>ConnectApi.CartSummary</td>
<td>Summary of the cart.</td>
<td>49.0</td>
</tr>
<tr>
<td>currentPageToken</td>
<td>String</td>
<td>Token identifying the current page.</td>
<td>49.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>49.0</td>
</tr>
<tr>
<td>hasErrors</td>
<td>Boolean</td>
<td>Specifies whether at least one of the results contains an error.</td>
<td>49.0</td>
</tr>
<tr>
<td>nextPageToken</td>
<td>String</td>
<td>Token identifying the next page, or null if there isn’t a next page.</td>
<td>49.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or null if there isn’t a next page.</td>
<td>49.0</td>
</tr>
</tbody>
</table>
### ConnectApi.CartItemProduct

Product summary for a cart item.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>fields</td>
<td>Map&lt;String, String&gt;</td>
<td>Map of product fields and values.</td>
<td>49.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the product.</td>
<td>49.0</td>
</tr>
<tr>
<td>productId</td>
<td>String</td>
<td>ID of the product.</td>
<td>49.0</td>
</tr>
<tr>
<td>productSubscriptionInformation</td>
<td></td>
<td>Reserved for future use.</td>
<td>59.0</td>
</tr>
<tr>
<td>purchaseQuantityRule</td>
<td>ConnectApi.PurchaseQuantityRule</td>
<td>If one exists, purchase quantity rule for the product.</td>
<td>52.0</td>
</tr>
<tr>
<td>sku</td>
<td>String</td>
<td>SKU of the product.</td>
<td>49.0</td>
</tr>
<tr>
<td>thumbnailImage</td>
<td>ConnectApi.ProductMedia</td>
<td>Thumbnail image of the product.</td>
<td>49.0</td>
</tr>
<tr>
<td>variationAttributes</td>
<td>Map&lt;String, ConnectApi.CartProductAttribute&gt;</td>
<td>Variation attributes associated with the product.</td>
<td>50.0</td>
</tr>
</tbody>
</table>

See also:
- ConnectApi.AbstractCartItem
- ConnectApi.WishlistItem

### ConnectApi.CartItemPromotionCollectionOutputRepresentation

Promotions for the items in a cart.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Currency code associated with the cart.</td>
<td>53.0</td>
</tr>
<tr>
<td>items</td>
<td>Map&lt;String, ConnectApi.&gt;</td>
<td>Collection of promotions.</td>
<td>52.0</td>
</tr>
</tbody>
</table>
### ConnectApi.CartItemResult

Result after requesting a cart item.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>cartItem</td>
<td>ConnectApi.AbstractCartItem</td>
<td>Cart item.</td>
<td>49.0</td>
</tr>
<tr>
<td>message</td>
<td>String</td>
<td>Message when the request isn’t successful.</td>
<td>49.0</td>
</tr>
<tr>
<td>status</td>
<td>String</td>
<td>Status for the request.</td>
<td>49.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.CartItemCollection
- ConnectApi.WishlistToCartResult

### ConnectApi.CartItemWithoutPrice

An item without price information in a cart.

Subclass of ConnectApi.AbstractCartItem.

No additional properties.

### ConnectApi.CartMessage

Cart message.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>message</td>
<td>String</td>
<td>Cart message.</td>
<td>49.0</td>
</tr>
<tr>
<td>messageId</td>
<td>String</td>
<td>ID of the object supplying the message.</td>
<td>49.0</td>
</tr>
<tr>
<td>relatedEntityId</td>
<td>String</td>
<td>ID of the entity, for example, cart, cart item, or cart tax, associated with the message.</td>
<td>49.0</td>
</tr>
<tr>
<td>severity</td>
<td>ConnectApi.CartMessageSeverity</td>
<td>Severity of cart message. Values are:</td>
<td>49.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Error</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Info</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Warning</td>
<td></td>
</tr>
</tbody>
</table>
### ConnectApi.CartMessagesSummary

Cart messages summary.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>errorCount</td>
<td>Integer</td>
<td>In <code>ConnectApi.CartItemResult</code>, the count of messages with the Error severity level.</td>
<td>49.0</td>
</tr>
<tr>
<td>hasErrors</td>
<td>Boolean</td>
<td>Specifies whether there are messages related to the entity (<code>true</code>) or not (<code>false</code>).</td>
<td>49.0</td>
</tr>
<tr>
<td>limitedMessages</td>
<td>List&lt;<code>ConnectApi.CartMessage</code>&gt;</td>
<td>In <code>ConnectApi.CartItemResult</code>, a limited list of messages related to the cart item. In <code>ConnectApi.CartSummary</code>, each message can be related to the cart or to another cart-related entity.</td>
<td>49.0</td>
</tr>
<tr>
<td>relatedEntityId</td>
<td>String</td>
<td>In <code>ConnectApi.CartItemResult</code>, the ID of the related cart item. In <code>ConnectApi.CartSummary</code>, each message can be related to the cart or to another cart-related entity.</td>
<td>49.0</td>
</tr>
<tr>
<td>totalLineItemsWithErrors</td>
<td>Integer</td>
<td>In <code>ConnectApi.CartItemResult</code>, either <code>null</code> if the cart item has no errors or 1 if the cart item has errors. In <code>ConnectApi.CartSummary</code>, the total number of product line items that contain errors.</td>
<td>50.0</td>
</tr>
</tbody>
</table>

SEE ALSO:  
`ConnectApi.AbstractCartItem`  
`ConnectApi.CartMessagesVisibilityResult`  

### ConnectApi.CartMessagesVisibilityResult

Result of setting the visibility for cart messages.
### ConnectApi.CartProductAttribute

Product attribute for a cart item.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>label</td>
<td>String</td>
<td>Label or display name of the attribute.</td>
<td>50.0</td>
</tr>
<tr>
<td>sequence</td>
<td>Integer</td>
<td>Sequence of the attribute within the attribute set.</td>
<td>50.0</td>
</tr>
<tr>
<td>value</td>
<td>String</td>
<td>Display value of the attribute.</td>
<td>50.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

- ConnectApi.CartItemProduct

### ConnectApi.CartPromotionCollection

All the promotions associated with the cart.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>cartId</td>
<td>String</td>
<td>ID of the cart.</td>
<td>53.0</td>
</tr>
<tr>
<td>cartStatus</td>
<td>ConnectApi.CartStatus</td>
<td>Status of the cart. Values are:</td>
<td>53.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>Active</strong>—Cart is created and available for modifications, like adding or removing products or promotions.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>Checkout</strong>—Cart is in checkout. If the customer modifies the cart, the current checkout session is canceled.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>Closed</strong>—Checkout is complete and an order was created. The cart cannot be modified.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>PendingClosed</strong>—Cart is marked to be closed, but the request isn’t completed yet. The cart can’t be modified. This value is available in API version 57.0 and later.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>PendingDelete</strong>—Cart is marked for delete, but the request isn’t completed yet. The cart can’t be modified.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• <strong>Processing</strong>—Cart is processing. For example, taxes are being calculated. The cart can’t be modified.</td>
<td></td>
</tr>
</tbody>
</table>
### ConnectApi.CartPromotionList
A list of promotions for a cart.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>promotions</td>
<td>List&lt;ConnectApi.CartPromotionList&gt;</td>
<td>Promotions associated with a cart.</td>
<td>54.0</td>
</tr>
</tbody>
</table>

### ConnectApi.CartPromotionOutputRepresentation
A promotion associated with a cart.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>adjustmentAmount</td>
<td>String</td>
<td>Adjustment amount out of the promotion.</td>
<td>53.0</td>
</tr>
<tr>
<td>couponCode</td>
<td>String</td>
<td>Coupon code for a promotion. A coupon code is available only for manual promotions, not for automatic promotions.</td>
<td>54.0</td>
</tr>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Currency ISO code associated with the cart.</td>
<td>57.0</td>
</tr>
<tr>
<td>displayName</td>
<td>String</td>
<td>Localized display name of the promotion.</td>
<td>52.0</td>
</tr>
<tr>
<td>promotionId</td>
<td>String</td>
<td>ID of the promotion.</td>
<td>53.0</td>
</tr>
</tbody>
</table>
| targetType | ConnectApi.CartPromotionType | Promotion target type. Values are:
- Cart—Promotion targets a cart.
- Item—Promotion targets an item in a cart. | 53.0 |
| termsAndConditions | String | Localized terms and conditions for the promotion. | 53.0 |

### ConnectApi.CartSummary
A cart summary.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>accountId</td>
<td>String</td>
<td>ID of the account for the cart.</td>
<td>49.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>---------------------</td>
<td>--------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>asyncOperationStatus</td>
<td>String</td>
<td>Asynchronous processing status of the cart, if asynchronous processing is enabled for the store. This property returns Completed in Apex, because Apex operations always run synchronously.</td>
<td>59.0</td>
</tr>
<tr>
<td>cartId</td>
<td>String</td>
<td>ID of the cart.</td>
<td>49.0</td>
</tr>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Three-letter ISO 4217 currency code associated with the cart.</td>
<td>49.0</td>
</tr>
<tr>
<td>grandTotalAmount</td>
<td>String</td>
<td>Grand total amount including shipping and tax for items in the cart, in the currency of the cart.</td>
<td>49.0</td>
</tr>
<tr>
<td>isSecondary</td>
<td>Boolean</td>
<td>Specifies whether the cart is secondary (true) or not (false).</td>
<td>53.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the cart.</td>
<td>49.0</td>
</tr>
<tr>
<td>ownerId</td>
<td>String</td>
<td>ID of the owner of the cart.</td>
<td>49.0</td>
</tr>
<tr>
<td>ownerOrderId</td>
<td>String</td>
<td>ID of the owner of the order.</td>
<td>58.0</td>
</tr>
<tr>
<td>purchaseOrderNumber</td>
<td>String</td>
<td>Purchase order for the cart.</td>
<td>50.0</td>
</tr>
<tr>
<td>status</td>
<td>ConnectApi.CartStatus</td>
<td>Status of the cart. Values are:</td>
<td>49.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Active—Cart is created and available for modifications, like adding or removing products or promotions.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Checkout—Cart is in checkout. If the customer modifies the cart, the current checkout session is canceled.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Closed—Checkout is complete and an order was created. The cart cannot be modified.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• PendingClosed—Cart is marked to be closed, but the request isn’t completed yet. The cart can’t be modified. This value is available in API version 57.0 and later.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• PendingDelete—Cart is marked for delete, but the request isn’t completed yet. The cart can’t be modified.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Processing—Cart is processing. For example, taxes are being calculated. The cart can’t be modified.</td>
<td></td>
</tr>
<tr>
<td>taxType</td>
<td>String</td>
<td>Tax type of the cart.</td>
<td>55.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Gross—Gross taxation policy.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Net—Net taxation policy.</td>
<td></td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>totalChargeAmount</td>
<td>String</td>
<td>Total amount for shipping and other charges in the currency of the cart.</td>
<td>49.0</td>
</tr>
<tr>
<td>totalListPrice</td>
<td>String</td>
<td>Total list price for the cart.</td>
<td>49.0</td>
</tr>
<tr>
<td>totalProductAmount</td>
<td>String</td>
<td>Total amount including discounts, but excluding shipping and tax, for product items in the cart.</td>
<td>49.0</td>
</tr>
<tr>
<td>totalProductAmountAfterAdjustments</td>
<td>String</td>
<td>Total product amount, including promotions.</td>
<td>52.0</td>
</tr>
<tr>
<td>totalProductCount</td>
<td>String</td>
<td>Total count of items in the cart.</td>
<td>49.0</td>
</tr>
<tr>
<td>totalProductListAmount</td>
<td>String</td>
<td>Total list amount for products in the cart.</td>
<td>59.0</td>
</tr>
<tr>
<td>totalPromotionalAdjustmentAmount</td>
<td>String</td>
<td>Total promotional adjustment amount for items in the cart.</td>
<td>52.0</td>
</tr>
<tr>
<td>totalTaxAmount</td>
<td>String</td>
<td>Total tax amount for the cart, including tax on shipping, if applicable.</td>
<td>49.0</td>
</tr>
</tbody>
</table>
| type                  | ConnectApi.CartType | Type of cart. Values are:  
• Cart—Cart created by a customer.  
• ReadOnly—Clone of a Template cart that the customer can check out with.  
• Template—Cart created by an internal user. | 49.0              |
| uniqueProductCount    | Integer       | Total count of unique items, or SKUs, in the cart.                          | 49.0              |
| webstoreId            | String        | ID of the webstore of the cart.                                             | 49.0              |

SEE ALSO:  
ConnectApi.CartItemCollection

**ConnectApi.CartToWishlistResult**

Result of copying products from a cart to a wishlist.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>productsAddedCount</td>
<td>Integer</td>
<td>Number of products copied from the cart to the wishlist.</td>
<td>50.0</td>
</tr>
<tr>
<td>wishlistId</td>
<td>String</td>
<td>ID of the wishlist that cart products were copied to.</td>
<td>50.0</td>
</tr>
</tbody>
</table>

**ConnectApi.CaseCommentCapability**

If a feed element has this capability, it has a case comment on the case feed.  
Subclass of ConnectApi.FeedElementCapability.
### Available Version

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>actorType</td>
<td><code>ConnectApi.CaseActorType</code></td>
<td>Specifies the type of user who made the comment.</td>
<td>32.0</td>
</tr>
<tr>
<td>createdBy</td>
<td><code>ConnectApi.Actor</code></td>
<td>Information about the user who created the comment.</td>
<td>32.0</td>
</tr>
<tr>
<td>createdDate</td>
<td><code>Datetime</code></td>
<td>ISO 8601 date string, for example, 2011-02-25T18:24:31.000Z.</td>
<td>32.0</td>
</tr>
<tr>
<td>eventType</td>
<td><code>ConnectApi.CaseCommentEventType</code></td>
<td>Specifies an event type for a comment in the case feed.</td>
<td>32.0</td>
</tr>
<tr>
<td>id</td>
<td><code>String</code></td>
<td>18-character ID of case comment.</td>
<td>32.0</td>
</tr>
<tr>
<td>published</td>
<td><code>Boolean</code></td>
<td>Specifies whether the comment has been published.</td>
<td>32.0</td>
</tr>
<tr>
<td>text</td>
<td><code>String</code></td>
<td>Text of the case comment.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

### ConnectApi.CdpActionResponse

Customer Data Platform action response.

This class is abstract.


<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>errors</td>
<td><code>List&lt;ConnectApi.CdpErrorResponse&gt;</code></td>
<td>List of errors that resulted from the action.</td>
<td>57.0</td>
</tr>
<tr>
<td>success</td>
<td><code>Boolean</code></td>
<td>Indicates whether the call was successful (<code>true</code>) or not (<code>false</code>).</td>
<td>57.0</td>
</tr>
</tbody>
</table>

### ConnectApi.CdpCalculatedInsightPage

Collection of calculated insights.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>collection</td>
<td><code>ConnectApi.CdpCalculatedInsightPageData</code></td>
<td>Collection of calculated insights.</td>
<td>57.0</td>
</tr>
</tbody>
</table>
ConnectApi.CdpCalculatedInsightPageData

Calculated insight collection data.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>count</td>
<td>Integer</td>
<td>Number of results returned in the page.</td>
<td>57.0</td>
</tr>
<tr>
<td>currentPageToken</td>
<td>String</td>
<td>Token identifying the current page.</td>
<td>57.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>57.0</td>
</tr>
<tr>
<td>items</td>
<td>List</td>
<td>List of calculated insights.</td>
<td>57.0</td>
</tr>
<tr>
<td>nextPageToken</td>
<td>String</td>
<td>Token identifying the next page, or null if there isn't a next page.</td>
<td>57.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or null if there isn't a next page.</td>
<td>57.0</td>
</tr>
<tr>
<td>previousPageToken</td>
<td>String</td>
<td>Token identifying the previous page, or null if there isn't a previous page.</td>
<td>57.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the previous page, or null if there isn't a previous page.</td>
<td>57.0</td>
</tr>
<tr>
<td>total</td>
<td>Integer</td>
<td>Total row count of calculated insights.</td>
<td>57.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.CdpCalculatedInsightPage

ConnectApi.CdpCalculatedInsightDataSource

Calculated insight data source.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>sourceApiName</td>
<td>String</td>
<td>Data source API name.</td>
<td>57.0</td>
</tr>
<tr>
<td>type</td>
<td>String</td>
<td>Data source type.</td>
<td>57.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.CdpCalculatedInsightDimension
- ConnectApi.CdpCalculatedInsightMeasure

ConnectApi.CdpCalculatedInsightDimension

Calculated insight dimension.
### Available Version

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>apiName</td>
<td>String</td>
<td>API name of the dimension.</td>
<td>57.0</td>
</tr>
<tr>
<td>creationType</td>
<td>String</td>
<td>Creation type of the dimension.</td>
<td>57.0</td>
</tr>
<tr>
<td>dataSource</td>
<td>ConnectApi.CdpCalculatedInsightDataSource</td>
<td>Data source of the dimension.</td>
<td>57.0</td>
</tr>
<tr>
<td>dataType</td>
<td>String</td>
<td>Data type of the dimension.</td>
<td>57.0</td>
</tr>
<tr>
<td>dateGranularity</td>
<td>String</td>
<td>Date granularity of the dimension.</td>
<td>57.0</td>
</tr>
<tr>
<td>displayName</td>
<td>String</td>
<td>Display name of the dimension.</td>
<td>57.0</td>
</tr>
<tr>
<td>fieldRole</td>
<td>String</td>
<td>Field role of the dimension.</td>
<td>57.0</td>
</tr>
<tr>
<td>formula</td>
<td>String</td>
<td>Formula of the dimension.</td>
<td>57.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.CdpCalculatedInsightOutput

### ConnectApi.CdpCalculatedInsightMeasure

Calculated insight measure.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>apiName</td>
<td>String</td>
<td>API name of the measure.</td>
<td>57.0</td>
</tr>
<tr>
<td>creationType</td>
<td>String</td>
<td>Creation type of the measure.</td>
<td>57.0</td>
</tr>
<tr>
<td>dataSource</td>
<td>ConnectApi.CdpCalculatedInsightDataSource</td>
<td>Data source of the measure.</td>
<td>57.0</td>
</tr>
<tr>
<td>dataType</td>
<td>String</td>
<td>Data type of the measure.</td>
<td>57.0</td>
</tr>
<tr>
<td>displayName</td>
<td>String</td>
<td>Display name of the measure.</td>
<td>57.0</td>
</tr>
<tr>
<td>fieldAggregationType</td>
<td>String</td>
<td>Field aggregation type of the measure.</td>
<td>57.0</td>
</tr>
<tr>
<td>fieldRole</td>
<td>String</td>
<td>Field role of the measure.</td>
<td>57.0</td>
</tr>
<tr>
<td>formula</td>
<td>String</td>
<td>Formula of the measure.</td>
<td>57.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.CdpCalculatedInsightOutput

### ConnectApi.CdpCalculatedInsightOutput

Calculated insight.
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>apiName</td>
<td>String</td>
<td>API name of the calculated insight.</td>
<td>57.0</td>
</tr>
<tr>
<td>calculatedInsight Status</td>
<td>String</td>
<td>Status of the calculated insight.</td>
<td>57.0</td>
</tr>
<tr>
<td>creationType</td>
<td>String</td>
<td>Creation type of the calculated insight.</td>
<td>57.0</td>
</tr>
<tr>
<td>dataSpace</td>
<td>String</td>
<td>Data space of the calculated insight.</td>
<td>57.0</td>
</tr>
<tr>
<td>definitionStatus</td>
<td>String</td>
<td>Definition status of the calculated insight.</td>
<td>57.0</td>
</tr>
<tr>
<td>definitionType</td>
<td>String</td>
<td>Definition type of the calculated insight.</td>
<td>57.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the calculated insight.</td>
<td>57.0</td>
</tr>
<tr>
<td>dimensions</td>
<td>List&lt;ConnectApi.CdpCalculatedInsightDimension&gt;</td>
<td>Dimensions of the calculated insight.</td>
<td>57.0</td>
</tr>
<tr>
<td>displayName</td>
<td>String</td>
<td>Display name of the calculated insight.</td>
<td>57.0</td>
</tr>
<tr>
<td>expression</td>
<td>String</td>
<td>Expression of the calculated insight.</td>
<td>57.0</td>
</tr>
<tr>
<td>isEnabled</td>
<td>Boolean</td>
<td>Specifies whether the calculated insight is enabled (true) or not (false).</td>
<td>57.0</td>
</tr>
<tr>
<td>lastCalcInsight StatusDateTime</td>
<td>String</td>
<td>Last calculated insight status date and time.</td>
<td>57.0</td>
</tr>
<tr>
<td>lastCalcInsight StatusErrorCode</td>
<td>String</td>
<td>Last calculated insight status error code.</td>
<td>57.0</td>
</tr>
<tr>
<td>lastRunDateTime</td>
<td>String</td>
<td>Last run date and time of the calculated insight.</td>
<td>57.0</td>
</tr>
<tr>
<td>lastRunStatus</td>
<td>String</td>
<td>Last run status of the calculated insight.</td>
<td>57.0</td>
</tr>
<tr>
<td>lastRunStatus DateTime</td>
<td>String</td>
<td>Last run status date and time of the calculated insight.</td>
<td>57.0</td>
</tr>
<tr>
<td>lastRunStatus ErrorCode</td>
<td>String</td>
<td>Last run status error code of the calculated insight.</td>
<td>57.0</td>
</tr>
<tr>
<td>measures</td>
<td>List&lt;ConnectApi.CdpCalculatedInsightMeasure&gt;</td>
<td>Measures of the calculated insight.</td>
<td>57.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.CdpCalculatedInsightPageData
- ConnectApi.CdpCalculatedInsightStandardActionResponseRepresentation

**ConnectApi.CdpCalculatedInsightStandardActionResponseRepresentation**

Response of the calculated insight run action.

ConnectApi.CdpErrorResponse

Error response.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>errorCode</td>
<td>String</td>
<td>Error code.</td>
<td>57.0</td>
</tr>
<tr>
<td>message</td>
<td>String</td>
<td>Message stating the reason for the error, if any.</td>
<td>57.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

ConnectApi.CdpActionResponse

ConnectApi.CdpIdentityResolutionMatchCriterionOutput

Identity resolution ruleset's match rule criterion.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>caseSensitiveMatch</td>
<td>Boolean</td>
<td>Specifies whether the criterion match is case sensitive (true) or not (false). Available only when matching is based on the party identifier.</td>
<td>58.0</td>
</tr>
<tr>
<td>entityName</td>
<td>String</td>
<td>API name of the Data Model Object the match rule applies to.</td>
<td>57.0</td>
</tr>
<tr>
<td>fieldName</td>
<td>String</td>
<td>Name of the field the criterion applies to.</td>
<td>57.0</td>
</tr>
</tbody>
</table>
| matchMethodType                | ConnectApi.CdpIdentityResolutionMatchMethodType | Match method for a match rule criterion. Values are:  
  • Exact—Exact match.  
  • ExactNormalized—Exact normalized match.  
  • Fuzzy—Fuzzy match with medium precision.  
  • FuzzyHigh—Fuzzy match with high precision.  
  • FuzzyLow—Fuzzy match with low precision. | 57.0              |
| partyIdentificationInfo        | ConnectApi.CdpIdentityResolutionMatchCriterionPartyIdentificationInfoOutput | Party Identifier information.                                             | 57.0              |
| shouldMatchOnBlank            | Boolean             | Specifies whether blank fields can be used for matching (true) or not (false). | 57.0              |

SEE ALSO:

ConnectApi.CdpIdentityResolutionMatchRuleOutput
ConnectApi.CdpIdentityResolutionMatchCriterionPartyIdentificationInfoOutput

Information when party identification is used in an identity resolution ruleset’s match rule criterion.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>partyName</td>
<td>String</td>
<td>Party identification name.</td>
<td>57.0</td>
</tr>
<tr>
<td>partyType</td>
<td>String</td>
<td>Party identification type.</td>
<td>57.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.CdpIdentityResolutionMatchCriterionOutput

ConnectApi.CdpIdentityResolutionMatchRuleOutput

Identity resolution ruleset’s match rule.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>criteria</td>
<td>List&lt;ConnectApi.CdpIdentityResolutionMatchCriterionOutput&gt;</td>
<td>Object and field the match rule applies to and the match method applied.</td>
<td>57.0</td>
</tr>
<tr>
<td>label</td>
<td>String</td>
<td>User friendly name for the identity resolution match rule.</td>
<td>57.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.CdpIdentityResolutionOutput

ConnectApi.CdpIdentityResolutionOutput

Identity resolution ruleset.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>anonymousUnifiedProfiles</td>
<td>Long</td>
<td>Count of anonymous unified profiles created by running the identity resolution ruleset.</td>
<td>57.0</td>
</tr>
<tr>
<td>configurationType</td>
<td>ConnectApi.CdpIdentityResolutionConfigurationType</td>
<td>Source object for an identity resolution ruleset. Values are:</td>
<td>57.0</td>
</tr>
<tr>
<td>consolidationRate</td>
<td>Double</td>
<td>Consolidation rate resulting from the run of an identity resolution ruleset.</td>
<td>57.0</td>
</tr>
<tr>
<td>dataSpaceName</td>
<td>String</td>
<td>Data space used as source data for an identity resolution ruleset.</td>
<td>57.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the identity resolution ruleset.</td>
<td>57.0</td>
</tr>
<tr>
<td>doesRun</td>
<td>Boolean</td>
<td>Specifies whether automatic job run scheduling is enabled for the ruleset</td>
<td>57.0</td>
</tr>
<tr>
<td>Automatically</td>
<td></td>
<td>(true) or not (false). If unspecified, defaults to false.</td>
<td></td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>Identity resolution ruleset's ID. This is not the identity resolution</td>
<td>57.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>resolution's extended ruleset ID (rulesetId).</td>
<td></td>
</tr>
<tr>
<td>knownUnifiedProfiles</td>
<td>Long</td>
<td>Count of known unified profiles created by running the identity resolution</td>
<td>57.0</td>
</tr>
<tr>
<td>label</td>
<td>String</td>
<td>User friendly name of the identity resolution ruleset.</td>
<td>57.0</td>
</tr>
<tr>
<td>lastJobCompleted</td>
<td>Datetime</td>
<td>Date and time the last job completed.</td>
<td>57.0</td>
</tr>
<tr>
<td>lastJobStatus</td>
<td>String</td>
<td>Last job's status. Possible values are:</td>
<td>57.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• SUCCESS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• IN_PROGRESS</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FAIL</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• SCHEDULED</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• SKIPPED</td>
<td></td>
</tr>
<tr>
<td>matchRules</td>
<td>List</td>
<td>List of match rules.</td>
<td>57.0</td>
</tr>
<tr>
<td></td>
<td>&lt;CdpIdentityResolution</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MatchRuleOutput&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>matchedSourceProfiles</td>
<td>Long</td>
<td>Count of matched source profiles identified by running the identity</td>
<td>57.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>resolution ruleset.</td>
<td></td>
</tr>
<tr>
<td>objectApiName</td>
<td>String</td>
<td>Object name of the identity resolution ruleset.</td>
<td>57.0</td>
</tr>
<tr>
<td>reconciliationRules</td>
<td>List</td>
<td>List of reconciliation rules.</td>
<td>57.0</td>
</tr>
<tr>
<td></td>
<td>&lt;CdpIdentityResolution</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ReconciliationRuleOutput&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>rulesetId</td>
<td>String</td>
<td>Extension ID of a ruleset. The ruleset ID must be unique and no longer than 4</td>
<td>57.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>characters. This ID is not the identifying ID for the ruleset.</td>
<td></td>
</tr>
<tr>
<td>rulesetStatus</td>
<td>String</td>
<td>Status of a ruleset job. Possible values are:</td>
<td>57.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NEW</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• PUBLISHING</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• PUBLISHED</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ERROR</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• DELETING</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• DELETE_FAILED</td>
<td></td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------</td>
<td>------------------------------------------------------------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>sourceProfiles</td>
<td>Long</td>
<td>Count of source profiles that were processed by a ruleset job.</td>
<td>57.0</td>
</tr>
<tr>
<td>totalUnifiedProfiles</td>
<td>Long</td>
<td>Count of unified profiles created by running the identity resolution ruleset.</td>
<td>57.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- `ConnectApi.CdpIdentityResolutionsOutput`

**ConnectApi.CdpIdentityResolutionReconciliationFieldRuleOutput**

Identity resolution ruleset's reconciliation rule for a field.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>fieldName</td>
<td>String</td>
<td>The field that this reconciliation rule applies to.</td>
<td>57.0</td>
</tr>
</tbody>
</table>
| ruleType            | `ConnectApi.CdpIdentityResolutionReconciliationRuleType` | Default reconciliation rule applied to fields in the object the reconciliation rule applies to. Values are:  
• LastUpdated  
• MostFrequent  
• SourceSequence  | 57.0             |
| shouldIgnoreEmptyValue | Boolean     | Specifies whether to ignore an empty value (`true`) or not (`false`).         | 57.0             |
| sources             | `List<ConnectApi.CdpIdentityResolutionReconciliationSourceOutput>` | If `ruleType` is `SourceSequence`, a prioritized list of data sources. | 57.0             |

**SEE ALSO:**
- `ConnectApi.CdpIdentityResolutionReconciliationRuleOutput`

**ConnectApi.CdpIdentityResolutionReconciliationRuleOutput**

Identity resolution ruleset's reconciliation rule for an object.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>entityName</td>
<td>String</td>
<td>API name of the Data Model Object the reconciliation rule applies to.</td>
<td>57.0</td>
</tr>
<tr>
<td>fields</td>
<td><code>List&lt;ConnectApi.CdpIdentityResolutionReconciliationFieldRuleOutput&gt;</code></td>
<td>Field-specific reconciliation rules that override this default rule for the specified field.</td>
<td>57.0</td>
</tr>
</tbody>
</table>
### Available Version Description Type Property Name

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>linkDmoName</td>
<td>String</td>
<td>API name of the unified link object created by the identity resolution process.</td>
<td>57.0</td>
</tr>
</tbody>
</table>
| ruleType            | ConnectApi.CdpIdentityResolution.ReconciliationRuleType | Default reconciliation rule applied to fields in the object the reconciliation rule applies to. Values are:  
  - LastUpdated  
  - MostFrequent  
  - SourceSequence | 57.0              |
| shouldIgnoreEmptyValue | Boolean               | Specifies whether to ignore an empty value (true) or not (false).           | 57.0              |
| sources             | List<ConnectApi.CdpIdentityResolution.ReconciliationSourceOutput> | If ruleType is SourceSequence, a list of data sources in priority order.     | 57.0              |
| unifiedDmoName      | String                    | API name of the unified data model object created by the identity resolution process. | 57.0              |

SEE ALSO:

- ConnectApi.CdpIdentityResolutionOutput

### ConnectApi.CdpIdentityResolutionReconciliationSourceOutput

Source for an identity resolution default reconciliation rule or field-specific rule using the SourceSequence match method.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>If the ruleType for a reconciliation rule is SourceSequence, API name of a source Data Lake Object.</td>
<td>57.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

- ConnectApi.CdpIdentityResolutionReconciliationRuleOutput
- ConnectApi.CdpIdentityResolutionReconciliationFieldRuleOutput

### ConnectApi.CdpIdentityResolutionRunNowOutput

Identity resolution ruleset run now output.
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>resultCode</td>
<td>ConnectApi.CdpIdentityResolutionRunNowResultCode</td>
<td>Result of an identity resolution ruleset job run. Values are:</td>
<td>57.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ExceededMaximumNumberOfSuccessfulRunsAllowedIn24Hours</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• IdentityResolutionJobIsAlreadyRunning</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NoPendingChangesJobRunSkipped</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• SuccessfullySubmittedIdentityResolutionJobRunRequest</td>
<td></td>
</tr>
</tbody>
</table>

**ConnectApi.CdpIdentityResolutionsOutput**

Identity resolution rulesets.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>identityResolutions</td>
<td>List&lt;ConnectApi.CdpIdentityResolutionOutput&gt;</td>
<td>List of identity resolution rulesets.</td>
<td>57.0</td>
</tr>
</tbody>
</table>

**ConnectApi.CdpQueryDataOutput**

Query data output.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>List&lt;Object&gt;</td>
<td>Result data set.</td>
<td>54.0</td>
</tr>
</tbody>
</table>

**ConnectApi.CdpQueryMetadataItem**

Metadata item.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>placeInOrder</td>
<td>Integer</td>
<td>Attribute place order in the result.</td>
<td>55.0</td>
</tr>
<tr>
<td>type</td>
<td>String</td>
<td>Metadata type for column.</td>
<td>55.0</td>
</tr>
<tr>
<td>typeCode</td>
<td>Integer</td>
<td>Metadata type code.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.CdpQueryOutputV2
### ConnectApi.CdpQueryMetadataOutput
Query metadata result.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>metadata</td>
<td>List&lt;Object&gt;</td>
<td>Metadata set.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

### ConnectApi.CdpQueryOutput
Query result.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>List&lt;Object&gt;</td>
<td>Result data set.</td>
<td>52.0</td>
</tr>
<tr>
<td>done</td>
<td>Boolean</td>
<td>Specifies whether the query is done (true) or not (false).</td>
<td>52.0</td>
</tr>
<tr>
<td>endTime</td>
<td>String</td>
<td>Query end time.</td>
<td>52.0</td>
</tr>
<tr>
<td>metadata</td>
<td>Map&lt;String, Object&gt;</td>
<td>Result metadata set.</td>
<td>52.0</td>
</tr>
<tr>
<td>queryId</td>
<td>String</td>
<td>Query ID.</td>
<td>52.0</td>
</tr>
<tr>
<td>rowCount</td>
<td>Integer</td>
<td>Number of rows in the result data set.</td>
<td>52.0</td>
</tr>
<tr>
<td>startTime</td>
<td>String</td>
<td>Query start time.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

### ConnectApi.CdpQueryOutputV2
Query output for the V2 API.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>List&lt;ConnectApi.CdpQueryV2Row&gt; (in version 55.0 and later)</td>
<td>Result data set.</td>
<td>54.0</td>
</tr>
<tr>
<td></td>
<td>List&lt;Object&gt; (version 54.0 only)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>done</td>
<td>Boolean</td>
<td>Specifies whether the query is done (true) or not (false).</td>
<td>54.0</td>
</tr>
<tr>
<td>endTime</td>
<td>String</td>
<td>Query end time.</td>
<td>54.0</td>
</tr>
<tr>
<td>metadata</td>
<td>Map&lt;String, ConnectApi.CdpQueryMetadataItem&gt; (version 55.0 and later)</td>
<td>Result metadata set.</td>
<td>54.0</td>
</tr>
</tbody>
</table>
**Available Version** | **Description** | **Type** | **Property Name**
---|---|---|---
54.0 | Next batch ID. | String | nextBatchId
54.0 | Use this property as the `nextBatchId` parameter in the `nextBatchAnsiSqlV2(nextBatchId)` method to get the next batch of data. | Map<String, Object> (version 54.0 only) |
54.0 | Query ID. | String | queryId
54.0 | Number of rows in the result data set. | Integer | rowCount
54.0 | Query start time. | String | startTime

**ConnectApi.CdpQueryV2Row**

Row in the query output for the V2 API.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>rowData</td>
<td>List&lt;Object&gt;</td>
<td>Row values.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

**See Also:**

*ConnectApi.CdpQueryOutputV2*

**ConnectApi.CdpSegmentActionOutput**

Segment action.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>errorCode</td>
<td>String</td>
<td>Error code associated with the action, if any.</td>
<td>57.0</td>
</tr>
<tr>
<td>errorMessage</td>
<td>String</td>
<td>Error message associated with the action, if any.</td>
<td>57.0</td>
</tr>
<tr>
<td>jobId</td>
<td>String</td>
<td>Job ID for the publish job.</td>
<td>56.0</td>
</tr>
<tr>
<td>partitionId</td>
<td>String</td>
<td>ID of the partition.</td>
<td>56.0</td>
</tr>
<tr>
<td>publishStatus</td>
<td>String</td>
<td>Publish status of the segment.</td>
<td>57.0</td>
</tr>
<tr>
<td>segmentId</td>
<td>String</td>
<td>ID of the segment.</td>
<td>56.0</td>
</tr>
</tbody>
</table>

**ConnectApi.CdpSegmentContainerOutput**

Segment container.
**ConnectApi.CdpSegmentDbtModel**
Segment dbt model.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>Dbt model name.</td>
<td>55.0</td>
</tr>
<tr>
<td>sql</td>
<td>String</td>
<td>Dbt model SQL.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.CdpSegmentDbtPipeline

**ConnectApi.CdpSegmentDbtPipeline**
Segment dbt pipeline.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>models</td>
<td>List&lt;ConnectApi.CdpSegmentDbtModel&gt;</td>
<td>Dbt models.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.CdpSegmentOutput

**ConnectApi.CdpSegmentMemberOutput**
Data Cloud segment member output.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>List&lt;ConnectApi.CdpSegmentMemberRowOutput&gt;</td>
<td>Result data set.</td>
<td>58.0</td>
</tr>
<tr>
<td>endTime</td>
<td>Datetime</td>
<td>Query end time.</td>
<td>58.0</td>
</tr>
</tbody>
</table>
### ConnectApi.CdpSegmentMemberRowOutput

Data Cloud segment member row output.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>deltaType</td>
<td>String</td>
<td>Delta type, for example, new, existing, or removed.</td>
<td>58.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>Segment member ID.</td>
<td>58.0</td>
</tr>
<tr>
<td>kqId</td>
<td>String</td>
<td>Fully qualified key ID.</td>
<td>58.0</td>
</tr>
<tr>
<td>snapshotType</td>
<td>String</td>
<td>Type of snapshot, for example, full or incremental.</td>
<td>58.0</td>
</tr>
<tr>
<td>timestamp</td>
<td>String</td>
<td>Timestamp.</td>
<td>58.0</td>
</tr>
<tr>
<td>versionStamp</td>
<td>String</td>
<td>Version timestamp.</td>
<td>58.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.CdpSegmentMemberOutput

### ConnectApi.CdpSegmentMembershipTableOutput

Data Cloud segment membership table.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>historyTable</td>
<td>String</td>
<td>Segment membership history table.</td>
<td>58.0</td>
</tr>
<tr>
<td>latestTable</td>
<td>String</td>
<td>Segment membership latest table.</td>
<td>58.0</td>
</tr>
</tbody>
</table>
### Segment Membership Profile Table

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>profileTable</td>
<td>String</td>
<td>Segment membership profile table.</td>
<td>58.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

ConnectApi.CdpSegmentOutput

### ConnectApi.CdpSegmentOutput

**Segment.**

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>apiName</td>
<td>String</td>
<td>API name of the segment.</td>
<td>56.0</td>
</tr>
<tr>
<td>dataSpace</td>
<td>String</td>
<td>Data space of the segment.</td>
<td>57.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Segment description.</td>
<td>55.0</td>
</tr>
<tr>
<td>developerName</td>
<td>String</td>
<td>Segment developer name.</td>
<td>55.0</td>
</tr>
<tr>
<td>displayName</td>
<td>String</td>
<td>Segment display name.</td>
<td>57.0</td>
</tr>
<tr>
<td>excludeCriteria</td>
<td>String</td>
<td>Segment exclude criteria.</td>
<td>57.0</td>
</tr>
<tr>
<td>includeCriteria</td>
<td>String</td>
<td>Segment include criteria.</td>
<td>57.0</td>
</tr>
<tr>
<td>includeDbt</td>
<td>ConnectApi.CdpSegmentDbtPipeline</td>
<td>Segment dbt pipeline.</td>
<td>55.0</td>
</tr>
<tr>
<td>lookalikeCriteria</td>
<td>Reserved for future use.</td>
<td></td>
<td>56.0</td>
</tr>
<tr>
<td>marketSegmentDefinitionId</td>
<td>String</td>
<td>ID of the market segment definition.</td>
<td>55.0</td>
</tr>
<tr>
<td>marketSegmentId</td>
<td>String</td>
<td>ID of the market segment.</td>
<td>56.0</td>
</tr>
<tr>
<td>nextPublishDateTime</td>
<td>String</td>
<td>Date and time of the next segment publish.</td>
<td>57.0</td>
</tr>
<tr>
<td>publishInterval</td>
<td>String</td>
<td>Segment publish interval.</td>
<td>55.0</td>
</tr>
<tr>
<td>publishScheduleEndDate</td>
<td>String</td>
<td>Publish schedule end date.</td>
<td>55.0</td>
</tr>
<tr>
<td>publishScheduleStartDateTime</td>
<td>String</td>
<td>Publish schedule start date time.</td>
<td>55.0</td>
</tr>
<tr>
<td>publishStatus</td>
<td>String</td>
<td>Segment publish status.</td>
<td>55.0</td>
</tr>
<tr>
<td>segmentMembershipDmo</td>
<td>ConnectApi.CdpSegmentMembershipTableOutput</td>
<td>Segment membership tables.</td>
<td>58.0</td>
</tr>
</tbody>
</table>
### ConnectApi.ChangeltemOutputRepresentation

The financial changes resulting from a change to one or more OrderItemSummaries. Most of the values represent the deltas of the values on the associated OrderSummary. The sign of each value is the opposite of the corresponding value on a change order record. For example, a discount is a positive value here and a negative value on a change order record.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>grandTotalAmount</td>
<td>Double</td>
<td>Change to the GrandTotalAmount field.</td>
<td>48.0</td>
</tr>
<tr>
<td>totalAdjDeliveryAmtWithTax</td>
<td>Double</td>
<td>Change to the TotalAdjDeliveryAmtWithTax field.</td>
<td>49.0</td>
</tr>
<tr>
<td>totalAdjDistAmountWithTax</td>
<td>Double</td>
<td>Change to the TotalAdjDistAmountWithTax field.</td>
<td>49.0</td>
</tr>
<tr>
<td>totalAdjProductAmtWithTax</td>
<td>Double</td>
<td>Change to the TotalAdjProductAmtWithTax field.</td>
<td>49.0</td>
</tr>
<tr>
<td>totalAdjustedDeliveryAmount</td>
<td>Double</td>
<td>Change to the TotalAdjustedDeliveryAmount field.</td>
<td>48.0</td>
</tr>
<tr>
<td>totalAdjustedDeliveryTaxAmount</td>
<td>Double</td>
<td>Change to the TotalAdjustedDeliveryTaxAmount field.</td>
<td>48.0</td>
</tr>
<tr>
<td>totalAdjustedProductAmount</td>
<td>Double</td>
<td>Change to the TotalAdjustedProductAmount field.</td>
<td>48.0</td>
</tr>
<tr>
<td>totalAdjustedProductTaxAmount</td>
<td>Double</td>
<td>Change to the TotalAdjustedProductTaxAmount field.</td>
<td>48.0</td>
</tr>
<tr>
<td>totalAdjustmentDistributedAmount</td>
<td>Double</td>
<td>Change to the TotalAdjustmentDistributedAmount field.</td>
<td>48.0</td>
</tr>
<tr>
<td>totalAdjustmentDistributedTaxAmount</td>
<td>Double</td>
<td>Change to the TotalAdjustmentDistributedTaxAmount field.</td>
<td>48.0</td>
</tr>
<tr>
<td>totalAmount</td>
<td>Double</td>
<td>Change to the TotalAmount field.</td>
<td>48.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>-----------------------</td>
<td>----------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>totalExcessFundsAmount</td>
<td>Double</td>
<td>Amount of excess funds available on the OrderPaymentSummaries related to the OrderSummary. It is equal to the captured amount that is owed as a refund but is not associated with an invoice or credit memo. Excess funds normally occur when order items are canceled before fulfillment but after payment has been captured. This situation is not common in the US, where funds are normally authorized but not captured until the fulfillment process begins. This value includes all current excess funds related to the OrderSummary, not only the funds related to the current change.</td>
<td>48.0</td>
</tr>
<tr>
<td>totalFeeAmount</td>
<td>Double</td>
<td>Total amount of the fees charged for the change.</td>
<td>57.0</td>
</tr>
<tr>
<td>totalFeeTaxAmount</td>
<td>Double</td>
<td>Total amount of tax on the fees charged for the change.</td>
<td>57.0</td>
</tr>
<tr>
<td>totalRefundableAmount</td>
<td>Double</td>
<td>Total amount available to be refunded. It is the sum of the excess funds and any outstanding change order grand total amounts that apply to post-fulfillment changes. This value includes all current refundable amounts related to the OrderSummary, not only the amount related to the current change.</td>
<td>48.0</td>
</tr>
<tr>
<td>totalRequiredFundsAmount</td>
<td>Double</td>
<td>The required funds associated with added order items.</td>
<td>54.0</td>
</tr>
<tr>
<td>totalTaxAmount</td>
<td>Double</td>
<td>Change to the TotalTaxAmount field.</td>
<td>48.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

- `ConnectApi.PreviewCancelOutputRepresentation`
- `ConnectApi.PreviewReturnOutputRepresentation`
- `ConnectApi.SubmitCancelOutputRepresentation`
- `ConnectApi.SubmitReturnOutputRepresentation`

**ConnectApi.ChatterActivity**

Chatter activity.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>commentCount</td>
<td>Integer</td>
<td>Total number of comments in the org or site made by the user.</td>
<td>28.0</td>
</tr>
<tr>
<td>commentReceivedCount</td>
<td>Integer</td>
<td>Total number of comments in the org or site received by the user.</td>
<td>28.0</td>
</tr>
</tbody>
</table>
### ConnectApi.ChatterActivitySummary

Summary of Chatter activity.

Subclass of `ConnectApi.UserFeedEntityActivitySummary`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>commentCount</td>
<td>Integer</td>
<td>Total number of comments in the org or site made by the user.</td>
<td>42.0</td>
</tr>
<tr>
<td>commentReceivedCount</td>
<td>Integer</td>
<td>Total number of comments in the org or site received by the user.</td>
<td>42.0</td>
</tr>
<tr>
<td>likeReceivedCount</td>
<td>Integer</td>
<td>Total number of likes and upvotes (in version 45.0 and later) on posts and comments in the org or site received by the user.</td>
<td>42.0</td>
</tr>
<tr>
<td>postCount</td>
<td>Integer</td>
<td>Total number of posts in the org or site made by the user.</td>
<td>42.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ChatterConversation

Chatter conversation.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>conversationId</td>
<td>String</td>
<td>ID for the conversation.</td>
<td>29.0</td>
</tr>
<tr>
<td>conversationUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the conversation.</td>
<td>29.0</td>
</tr>
<tr>
<td>members</td>
<td>List&lt;ConnectApi.UserSummary&gt;</td>
<td>List of users in the conversation.</td>
<td>29.0</td>
</tr>
<tr>
<td>messages</td>
<td>ConnectApi.ChatterMessagePage</td>
<td>Content of the conversation.</td>
<td>29.0</td>
</tr>
<tr>
<td>read</td>
<td>Boolean</td>
<td>Specifies if the conversation is read (true) or not read (false)</td>
<td>29.0</td>
</tr>
</tbody>
</table>
ConnectApi.ChatterConversationPage

Chatter conversation page.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>conversations</td>
<td>List&lt;ConnectApi.ChatterConversationSummary&gt;</td>
<td>List of conversations on the page.</td>
<td>29.0</td>
</tr>
<tr>
<td>currentPageToken</td>
<td>String</td>
<td>Token identifying the current page.</td>
<td>29.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>29.0</td>
</tr>
<tr>
<td>nextPageToken</td>
<td>String</td>
<td>Token identifying the next page, or null if there isn't a next page.</td>
<td>29.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or null if there isn't a next page.</td>
<td>29.0</td>
</tr>
</tbody>
</table>

ConnectApi.ChatterConversationSummary

Chatter conversation summary.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>ID for the conversation summary.</td>
<td>29.0</td>
</tr>
<tr>
<td>latestMessage</td>
<td>ConnectApi.ChatterMessage</td>
<td>Contents of the latest message.</td>
<td>29.0</td>
</tr>
<tr>
<td>members</td>
<td>List&lt;ConnectApi.UserSummary&gt;</td>
<td>List of members in the conversation.</td>
<td>29.0</td>
</tr>
<tr>
<td>read</td>
<td>Boolean</td>
<td>Specifies if the conversation is read (true) or not read (false).</td>
<td>29.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>Connect REST API URL to the conversation summary.</td>
<td>29.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

ConnectApi.ChatterConversationPage

ConnectApi.ChatterGroup

Chatter group.

This class is abstract.

Subclass of ConnectApi.ActorWithId.

Superclass of:

• ConnectApi.ChatterGroupDetail
• ConnectApi.ChatterGroupSummary
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>additionalLabel</td>
<td>String</td>
<td>An extra label for the group, for example, “Archived,” “Private,” or “Private With Customers.” If there isn’t an extra label, the value is null.</td>
<td>30.0</td>
</tr>
<tr>
<td>announcement</td>
<td>ConnectApi.Announcement</td>
<td>The current announcement for this group. An announcement displays in a designated location in the Salesforce UI until 11:59 p.m. on its expiration date, unless it's deleted or replaced by another announcement.</td>
<td>31.0</td>
</tr>
<tr>
<td>bannerPhoto</td>
<td>ConnectApi.BannerPhoto</td>
<td>The banner photo for the group.</td>
<td>36.0</td>
</tr>
<tr>
<td>canHaveChatterGuests</td>
<td>Boolean</td>
<td>true if this group allows Chatter guests.</td>
<td>28.0</td>
</tr>
<tr>
<td>community</td>
<td>ConnectApi.Reference</td>
<td>Information about the Experience Cloud site the group is in.</td>
<td>28.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Group’s description.</td>
<td>28.0</td>
</tr>
<tr>
<td>emailToChatterAddress</td>
<td>String</td>
<td>Group’s email address for posting to the group by email. Returns null if Chatter emails and posting to Chatter by email aren’t both enabled in your organization.</td>
<td>30.0</td>
</tr>
<tr>
<td>isArchived</td>
<td>Boolean</td>
<td>Specifies whether the group is archived (true) or not (false).</td>
<td>29.0</td>
</tr>
<tr>
<td>isAutoArchiveDisabled</td>
<td>Boolean</td>
<td>Specifies whether automatic archiving is disabled for the group (true) or not (false).</td>
<td>29.0</td>
</tr>
<tr>
<td>isBroadcast</td>
<td>Boolean</td>
<td>Specifies whether the group is a broadcast group (true) or not (false). In a broadcast group, only group owners and managers can post to the group.</td>
<td>36.0</td>
</tr>
<tr>
<td>lastFeedElementPostDate</td>
<td>Datetime</td>
<td>ISO 8601 date string, for example, 2011-02-25T18:24:31.000Z, of the most recent feed element posted to the group.</td>
<td>31.0</td>
</tr>
<tr>
<td>lastFeedItemPostDate</td>
<td>Datetime</td>
<td>ISO 8601 date string, for example, 2011-02-25T18:24:31.000Z, of the most recent feed item posted to the group. Use lastFeedElementPosted.</td>
<td>28.0–30.0</td>
</tr>
<tr>
<td>memberCount</td>
<td>Integer</td>
<td>Total number of group members.</td>
<td>28.0</td>
</tr>
<tr>
<td>myRole</td>
<td>ConnectApi.GroupMembershipType</td>
<td>Type of membership the user has with the group.</td>
<td>28.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• GroupOwner</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• GroupManager</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NotAMember</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NotAMemberPrivateRequested</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• StandardMember</td>
<td></td>
</tr>
<tr>
<td>mySubscription</td>
<td>ConnectApi.Reference</td>
<td>If the context user is a member of this group, contains information about that subscription; otherwise, returns null.</td>
<td>28.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the group.</td>
<td>28.0</td>
</tr>
</tbody>
</table>
### ConnectApi.ChatterGroupDetail

Chatter group details.

Subclass of `ConnectApi.ChatterGroup`.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>owner</td>
<td><code>ConnectApi.UserSummary</code></td>
<td>Information about the owner of the group.</td>
<td>28.0</td>
</tr>
<tr>
<td>photo</td>
<td><code>ConnectApi.Photo</code></td>
<td>Information about the group photo.</td>
<td>28.0</td>
</tr>
<tr>
<td>visibility</td>
<td><code>ConnectApi.GroupVisibilityType</code></td>
<td>Group visibility type. Valid values are:</td>
<td>28.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- PrivateAccess—Only members of the group can see posts to this group.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- PublicAccess—All users within the Experience Cloud site can see posts to</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>this group.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Unlisted—Reserved for future use.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>fileCount</td>
<td><code>Integer</code></td>
<td>The number of files posted to the group.</td>
<td>28.0</td>
</tr>
<tr>
<td>information</td>
<td><code>ConnectApi.GroupInformation</code></td>
<td>Describes the Information section of the group. If the group is private,</td>
<td>28.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>this section is visible only to members. If the context user is not a</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>member of the group or does not have Modify All Data or View All Data</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>permission, this value is <code>null</code>.</td>
<td></td>
</tr>
<tr>
<td>pending Requests</td>
<td><code>Integer</code></td>
<td>The number of requests to join a group that are in a pending state.</td>
<td>29.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- `ConnectApi.ChatterGroupPage`
- `ConnectApi.UserGroupDetailPage`

### ConnectApi.ChatterGroupPage

Page of groups.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageUrl</td>
<td><code>String</code></td>
<td>Connect REST API URL identifying the current page.</td>
<td>28.0</td>
</tr>
<tr>
<td>groups</td>
<td><code>List&lt;ConnectApi.Chatter&gt;</code></td>
<td>List of group details.</td>
<td>28.0</td>
</tr>
</tbody>
</table>
ConnectApi.ChatterGroupSummary

Chatter group summary.
Subclass of ConnectApi.ChatterGroup.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>fileCount</td>
<td>Integer</td>
<td>The number of files posted to the group.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
  - ConnectApi.ChatterGroupSummaryPage
  - ConnectApi.UserGroupPage

ConnectApi.ChatterGroupSummaryPage

Page of group summaries.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>29.0</td>
</tr>
<tr>
<td>groups</td>
<td>List&lt;ConnectApi.ChatterGroupSummary&gt;</td>
<td>List of group summary objects.</td>
<td>29.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or null if there isn’t a next page.</td>
<td>29.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the previous page, or null if there isn’t a previous page.</td>
<td>29.0</td>
</tr>
</tbody>
</table>

ConnectApi.ChatterLike

Chatter like information.
## ConnectApi.ChatterLikePage

Page of Chatter likes.

### Available Version

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageToken</td>
<td>Integer</td>
<td>Token identifying the current page.</td>
<td>28.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>28.0</td>
</tr>
<tr>
<td>items</td>
<td>List&lt;ConnectApi.ChatterLike&gt;</td>
<td>List of likes.</td>
<td>32.0</td>
</tr>
<tr>
<td>likes</td>
<td>List&lt;ConnectApi.ChatterLike&gt;</td>
<td>List of likes.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>nextPageToken</td>
<td>Integer</td>
<td>Token identifying the next page, or null if there isn't a next page.</td>
<td>28.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or null if there isn't a next page.</td>
<td>28.0</td>
</tr>
<tr>
<td>previousPageToken</td>
<td>Integer</td>
<td>Token identifying the previous page, or null if there isn't a previous page.</td>
<td>28.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the previous page, or null if there isn't a previous page.</td>
<td>28.0</td>
</tr>
<tr>
<td>total</td>
<td>Integer</td>
<td>Total number of likes across all pages.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

### Important

As of API version 32.0, use the `items` property.

### SEE ALSO:

- ConnectApi.ChatterLikesCapability
- ConnectApi.Comment
**ConnectApi.ChatterLikesCapability**

If a feed element has this capability, the context user can like it. Exposes information about existing likes.

Subclass of `ConnectApi.FeedElementCapability`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isLikedByCurrentUser</td>
<td>Boolean</td>
<td>Indicates whether the feed element is liked by the context user (<code>true</code>) or not (<code>false</code>).</td>
<td>32.0</td>
</tr>
<tr>
<td>page</td>
<td>ConnectApi.ChatterLikePage</td>
<td>Likes information for this feed element.</td>
<td>32.0</td>
</tr>
<tr>
<td>likesMessage</td>
<td>ConnectApi.MessageBody</td>
<td>A message body that describes who likes the feed element.</td>
<td>32.0</td>
</tr>
<tr>
<td>myLike</td>
<td>ConnectApi.Reference</td>
<td>If the context user has liked the feed element, this property is a reference to the specific like, <code>null</code> otherwise.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- `ConnectApi.FeedElementCapabilities`

**ConnectApi.ChatterMessage**

Chatter message.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>body</td>
<td>ConnectApi.MessageBody</td>
<td>Contents of the message.</td>
<td>29.0</td>
</tr>
<tr>
<td>conversationId</td>
<td>String</td>
<td>ID for the conversation.</td>
<td>29.0</td>
</tr>
<tr>
<td>conversationUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the conversation.</td>
<td>29.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the message.</td>
<td>29.0</td>
</tr>
<tr>
<td>recipients</td>
<td>List&lt;ConnectApi.UserSummary&gt;</td>
<td>List of the recipients of the message.</td>
<td>29.0</td>
</tr>
<tr>
<td>sender</td>
<td>ConnectApi.UserSummary</td>
<td>Sender of the message.</td>
<td>29.0</td>
</tr>
<tr>
<td>sendingCommunity</td>
<td>ConnectApi.Reference</td>
<td>Information about the Experience Cloud site from which the message was sent.</td>
<td>32.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Returns <code>null</code> for the default site or if digital experiences isn’t enabled.</td>
<td></td>
</tr>
<tr>
<td>sentDate</td>
<td>Datetime</td>
<td>The date and time the message was sent.</td>
<td>29.0</td>
</tr>
</tbody>
</table>
url | String | Connect REST API URL identifying the current page of the conversation. | 29.0

SEE ALSO:
- ConnectApi.ChatterConversationSummary
- ConnectApi.ChatterMessagePage

**ConnectApi.ChatterMessagePage**

Chatter message page.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageToken</td>
<td>String</td>
<td>Token identifying the current page.</td>
<td>29.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>29.0</td>
</tr>
<tr>
<td>messages</td>
<td>List&lt;ConnectApi.ChatterMessage&gt;</td>
<td>Messages on the current page.</td>
<td>29.0</td>
</tr>
<tr>
<td>nextPageToken</td>
<td>String</td>
<td>Token identifying the next page, or null if there isn't a next page.</td>
<td>29.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or null if there isn't a next page.</td>
<td>29.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.ChatterConversation

**ConnectApi.ChatterStream**

A Chatter feed stream.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>community</td>
<td>ConnectApi.CommunitySummary</td>
<td>Experience Cloud site where the stream is.</td>
<td>41.0</td>
</tr>
<tr>
<td>createdDate</td>
<td>Datetime</td>
<td>Date the stream was created.</td>
<td>39.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the stream.</td>
<td>39.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>18-character ID of the stream.</td>
<td>39.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the stream.</td>
<td>39.0</td>
</tr>
<tr>
<td>subscriptions</td>
<td>List&lt;ConnectApi.FeedEnabledEntity&gt;</td>
<td>List of entities whose feeds are included in the stream.</td>
<td>39.0</td>
</tr>
</tbody>
</table>
### ConnectApi.ChatterStreamPage

A collection of Chatter feed streams.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>URL to the current page of streams.</td>
<td>39.0</td>
</tr>
<tr>
<td>items</td>
<td>List&lt;ConnectApi.ChatterStream&gt;</td>
<td>List of streams.</td>
<td>39.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>URL to the next page of streams.</td>
<td>39.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>In version 39.0, all streams are included in currentPageUrl and nextPageUrl is null.</td>
<td></td>
</tr>
<tr>
<td>total</td>
<td>Integer</td>
<td>Total number of streams in the collection.</td>
<td>39.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ClientInfo

Client information.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>applicationName</td>
<td>String</td>
<td>Name of the connected app used for authentication.</td>
<td>28.0</td>
</tr>
<tr>
<td>applicationUrl</td>
<td>String</td>
<td>Value from the Info URL field of the connected app used for authentication.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

### ConnectApi.CloseCapability

If a feed element has this capability, users with permission can close it.

Users can’t edit (specifically the feed item body or title), comment on, or delete a closed feed element. If the closed feed element is a poll, users can’t vote on it. Users can’t edit (specifically the comment body) or delete a comment on a closed feed element or select or remove it as best answer.
Admins and moderators can edit and delete closed feed elements and comments on closed feed elements. Admins and moderators can select or remove the best answer status on comments on closed feed elements.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>canContextUser</td>
<td>Boolean</td>
<td>Specifies whether the context user has permission to set the feed element to closed (true) or not (false).</td>
<td>43.0</td>
</tr>
<tr>
<td>UpdateIsClosed</td>
<td>Boolean</td>
<td></td>
<td></td>
</tr>
<tr>
<td>isClosed</td>
<td>Boolean</td>
<td>Specifies whether the feed element is closed (true) or not (false).</td>
<td>43.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.FeedElementCapabilities

### ConnectApi.Comment

A comment.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>attachment</td>
<td>ConnectApi.FeedItem.Attachment</td>
<td>If the comment contains an attachment, property value is ContentAttachment. If the comment does not contain an attachment, it is null.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>body</td>
<td>ConnectApi.FeedBody</td>
<td>Body of the comment.</td>
<td>28.0</td>
</tr>
<tr>
<td>capabilities</td>
<td>ConnectApi.CommentCapabilities</td>
<td>Capabilities associated with the comment, such as any file attachments.</td>
<td>32.0</td>
</tr>
<tr>
<td>clientInfo</td>
<td>ConnectApi.ClientInfo</td>
<td>Information about the connected app used to authenticate the connection.</td>
<td>28.0</td>
</tr>
<tr>
<td>createdDate</td>
<td>Datetime</td>
<td>ISO 8601 date string, for example, 2011-02-25T18:24:31.000Z.</td>
<td>28.0</td>
</tr>
<tr>
<td>feedElement</td>
<td>ConnectApi.Reference</td>
<td>Feed element on which the comment is posted.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>feedItem</td>
<td>ConnectApi.Reference</td>
<td>Feed item on which the comment is posted.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>Comment’s 18-character ID.</td>
<td>28.0</td>
</tr>
<tr>
<td>isDelete</td>
<td>Boolean</td>
<td>If this property is true, the context user can’t delete the comment. If this property is false, the context user might be able to delete the comment.</td>
<td>28.0</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>likes</td>
<td>ConnectApi.ChatterLikePage</td>
<td>The first page of likes for the comment. This property has no information for comments on direct messages.</td>
<td>28.0</td>
</tr>
<tr>
<td>likesMessage</td>
<td>ConnectApi.MessageBody</td>
<td>A message body that describes who likes the comment. This property is <code>null</code> for comments on direct messages</td>
<td>28.0</td>
</tr>
<tr>
<td>moderation</td>
<td>ConnectApi.ModerationFlags</td>
<td>Information about the moderation flags on a comment. If ConnectApi.Features.communityModeration is <code>false</code>, this property is <code>null</code>.</td>
<td>29.0</td>
</tr>
<tr>
<td>myLike</td>
<td>ConnectApi.Reference</td>
<td>If the context user liked the comment, this property is a reference to the specific like, <code>null</code> otherwise. This property is <code>null</code> for comments on direct messages.</td>
<td>28.0</td>
</tr>
<tr>
<td>parent</td>
<td>ConnectApi.Reference</td>
<td>Information about the parent feed-item for this comment.</td>
<td>28.0</td>
</tr>
<tr>
<td>relativeCreatedDate</td>
<td>String</td>
<td>The created date formatted as a relative, localized string, for example, “17m ago” or “Yesterday.”</td>
<td>28.0</td>
</tr>
<tr>
<td>threadLevel</td>
<td>Integer</td>
<td>Level of nesting for a comment. 0 indicates a standard comment with a parent post. 1 indicates a threaded comment with a parent comment and a parent post. 2 indicates a threaded comment with two parent comments and a parent post. The UI is limited to these three levels.</td>
<td>44.0</td>
</tr>
<tr>
<td>threadParentId</td>
<td>String</td>
<td>ID of the parent comment for a threaded comment.</td>
<td>44.0</td>
</tr>
<tr>
<td>type</td>
<td>ConnectApi.CommentType</td>
<td>Type of comment.</td>
<td>28.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ContentComment—Comment holds a content capability.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• TextComment—Comment contains only text.</td>
<td></td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>Connect REST API URL to this comment.</td>
<td>28.0</td>
</tr>
<tr>
<td>user</td>
<td>ConnectApi.UserSummary</td>
<td>Information about the comment author.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.CommentPage
- ConnectApi.QuestionAndAnswersCapability

**ConnectApi.CommentCapabilities**

A set of capabilities on a comment.
## Available Version

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>comments</td>
<td>ConnectApi.CommentsCapability</td>
<td>If a comment has this capability, it has threaded comments.</td>
<td>44.0</td>
</tr>
<tr>
<td>content</td>
<td>ConnectApi.ContentCapability</td>
<td>If a comment has this capability, it has a file attachment. Most ConnectApi.ContentCapability properties are null if the content has been deleted from the feed element or if the access has changed to private.</td>
<td>32.0</td>
</tr>
<tr>
<td>edit</td>
<td>ConnectApi.EditCapability</td>
<td>If a comment has this capability, users who have permission can edit it.</td>
<td>34.0</td>
</tr>
<tr>
<td>feedEntityShare</td>
<td>ConnectApi.FeedEntityShareCapability</td>
<td>If a comment has this capability, a feed entity is shared with it.</td>
<td>42.0</td>
</tr>
<tr>
<td>record</td>
<td>ConnectApi.RecordCapability</td>
<td>If a comment has this capability, it has a record attachment.</td>
<td>42.0</td>
</tr>
<tr>
<td>status</td>
<td>ConnectApi.StatusCapability</td>
<td>If a comment has this capability, it has a status that determines its visibility.</td>
<td>38.0</td>
</tr>
<tr>
<td>upDownVote</td>
<td>ConnectApi.UpDownVoteCapability</td>
<td>If a comment has this capability, users can upvote or downvote it.</td>
<td>41.0</td>
</tr>
<tr>
<td>verified</td>
<td>ConnectApi.VerifiedCapability</td>
<td>If a comment has this capability, users with permission can mark it as verified or unverified.</td>
<td>41.0</td>
</tr>
</tbody>
</table>

See also: [ConnectApi.Comment](#)

### ConnectApi.CommentPage

A page of comments.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>comments</td>
<td>List&lt;ConnectApi.Comment&gt;</td>
<td>Collection of comments.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td></td>
<td>Important:</td>
<td>As of version 32.0, use the <code>items</code> property.</td>
<td></td>
</tr>
<tr>
<td>currentPageToken</td>
<td>String</td>
<td>Token identifying the current page.</td>
<td>28.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>28.0</td>
</tr>
<tr>
<td>items</td>
<td>List&lt;ConnectApi.Comment&gt;</td>
<td>Collection of comments for this feed element.</td>
<td>32.0</td>
</tr>
</tbody>
</table>
### ConnectApi.CommentsCapability

Summary of the comment.

Subclass of `ConnectApi.UserActivitySummary`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>commentId</td>
<td>String</td>
<td>ID of the comment.</td>
<td>42.0</td>
</tr>
</tbody>
</table>

### ConnectApi.CommentsCapability

If a feed element or comment has this capability, the context user can add a comment to it.

Subclass of `ConnectApi.FeedElementCapability`. 

---

**Available Version**

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>nextPageToken</td>
<td>String</td>
<td>Token identifying the next page, or <code>null</code> if there isn’t a next page.</td>
<td>28.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or <code>null</code> if there isn’t a next page.</td>
<td>28.0</td>
</tr>
<tr>
<td>previousPageToken</td>
<td>String</td>
<td>Token identifying the previous page, or <code>null</code> if there isn’t a previous page.</td>
<td>44.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the previous page, or <code>null</code> if there isn’t a previous page.</td>
<td>44.0</td>
</tr>
<tr>
<td>total</td>
<td>Integer</td>
<td>Total number of published comments for the parent feed element.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

- `ConnectApi.CommentsCapability`
- `ConnectApi.CommentSummary`
### ConnectApi.CommentPage

The comments information for this feed element or comment. Threaded comments are supported in version 44.0 and later.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>page</td>
<td>ConnectApi.CommentPage</td>
<td>The comments information for this feed element or comment. Threaded comments are supported in version 44.0 and later.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.FeedElementCapabilities

### ConnectApi.CommerceActionResult

Result of executing a commerce action.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isSuccess</td>
<td>Boolean</td>
<td>Specifies whether the action is a success (true) or not (false).</td>
<td>53.0</td>
</tr>
<tr>
<td>message</td>
<td>String</td>
<td>Action result message.</td>
<td>53.0</td>
</tr>
</tbody>
</table>

### ConnectApi.CommerceAddressCollection

A collection of Commerce addresses.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>count</td>
<td>Integer</td>
<td>Count of addresses.</td>
<td>54.0</td>
</tr>
<tr>
<td>currentPageToken</td>
<td>String</td>
<td>Token to the current page of addresses.</td>
<td>54.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>URL to the current page of addresses.</td>
<td>54.0</td>
</tr>
<tr>
<td>items</td>
<td>AddressDetails</td>
<td>Address Details</td>
<td>54.0</td>
</tr>
<tr>
<td>nextPageToken</td>
<td>String</td>
<td>Token to the next page of addresses.</td>
<td>54.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>URL to the next page of addresses.</td>
<td>54.0</td>
</tr>
<tr>
<td>pageSize</td>
<td>Integer</td>
<td>Page size for addresses.</td>
<td>54.0</td>
</tr>
<tr>
<td>previousPageToken</td>
<td>String</td>
<td>Token to previous page of addresses.</td>
<td>54.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>URL to the previous page of addresses.</td>
<td>54.0</td>
</tr>
<tr>
<td>sortOrder</td>
<td>CommerceAddressSort</td>
<td>Sort order for Commerce addresses.</td>
<td>54.0</td>
</tr>
</tbody>
</table>
- CreatedDateAsc—Sort in ascending order of created date.
- CreatedDateDesc—Sort in descending order of created date.
- NameAsc—Sort in ascending order of name.
ConnectApi.CommerceAddressOutput

Address for a Commerce account.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>addressId</td>
<td>String</td>
<td>ID of the address.</td>
<td>54.0</td>
</tr>
<tr>
<td>addressType</td>
<td>String</td>
<td>Type of address (for example, &quot;Shipping&quot; or &quot;Billing&quot;).</td>
<td>54.0</td>
</tr>
<tr>
<td>city</td>
<td>String</td>
<td>The address city.</td>
<td>54.0</td>
</tr>
<tr>
<td>companyName</td>
<td>String</td>
<td>The address company name.</td>
<td>57.0</td>
</tr>
<tr>
<td>country</td>
<td>String</td>
<td>The address country.</td>
<td>54.0</td>
</tr>
<tr>
<td>countryCode</td>
<td>String</td>
<td>Two-character country code.</td>
<td>54.0–58.0</td>
</tr>
<tr>
<td>fields</td>
<td>Map&lt;String, RecordField&gt;</td>
<td>A list of custom address fields, if any.</td>
<td>54.0</td>
</tr>
<tr>
<td>firstName</td>
<td>String</td>
<td>The address first name.</td>
<td>57.0</td>
</tr>
<tr>
<td>isDefault</td>
<td>Boolean</td>
<td>Indicates whether a contact’s address is the preferred method of communication (true) or not (false). The default value is false.</td>
<td>54.0</td>
</tr>
<tr>
<td>lastName</td>
<td>String</td>
<td>The address last name.</td>
<td>57.0</td>
</tr>
<tr>
<td>middleName</td>
<td>String</td>
<td>The address middle name.</td>
<td>57.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the contact.</td>
<td>54.0</td>
</tr>
<tr>
<td>phoneNumber</td>
<td>String</td>
<td>The address phone number.</td>
<td>57.0</td>
</tr>
<tr>
<td>postalCode</td>
<td>String</td>
<td>Zip code or postal code for the address.</td>
<td>54.0</td>
</tr>
<tr>
<td>region</td>
<td>String</td>
<td>The address state.</td>
<td>54.0</td>
</tr>
<tr>
<td>regionCode</td>
<td>String</td>
<td>The address state code.</td>
<td>54.0–58.0</td>
</tr>
<tr>
<td>street</td>
<td>String</td>
<td>The address street.</td>
<td>54.0</td>
</tr>
</tbody>
</table>

ConnectApi.CommerceProductSearchResults

Product search results information.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>categories</td>
<td>ConnectApi. SearchCategory</td>
<td>Categories from the search results.</td>
<td>52.0</td>
</tr>
</tbody>
</table>
### ConnectApi.CommerceProductSellingModel

Product selling model information.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isSubscriptionProduct</td>
<td>Boolean</td>
<td>Indicates whether the product selling model is a subscription product or not.</td>
<td>59.0</td>
</tr>
</tbody>
</table>

### ConnectApi.CommerceProductSummary

Summary of a product in product search results.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>productSellingModelInformation</td>
<td>ConnectApi.CommerceProductSellingModel</td>
<td>Information about the product selling model.</td>
<td>59.0</td>
</tr>
<tr>
<td>purchaseQuantityRule</td>
<td>ConnectApi.PurchaseQuantityRule</td>
<td>Purchase quantity rule for the product.</td>
<td>58.0</td>
</tr>
<tr>
<td>variationAttributeSet</td>
<td>String</td>
<td>Reserved for future use.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

### ConnectApi.CommerceProductSummaryPage

Collection of product summary representations in product search results.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>pageSize</td>
<td>Integer</td>
<td>Number of products per page in search results.</td>
<td>55.0</td>
</tr>
<tr>
<td>products</td>
<td>ConnectApi.CommerceProductSummary</td>
<td>Collection of product summaries.</td>
<td>55.0</td>
</tr>
<tr>
<td>total</td>
<td>Long</td>
<td>Number of products in search results across all pages.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

### ConnectApi.CommerceSearchIndex

Index information.
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>completionDate</td>
<td>Datetime</td>
<td>Completion date and time of the index.</td>
<td>52.0</td>
</tr>
<tr>
<td>createdDate</td>
<td>Datetime</td>
<td>Creation date of the index.</td>
<td>52.0</td>
</tr>
<tr>
<td>creationType</td>
<td>ConnectApi. CommerceSearch IndexCreationType</td>
<td>Creation type of the index. Values are:</td>
<td>52.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Manual</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Scheduled</td>
<td></td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the index.</td>
<td>52.0</td>
</tr>
<tr>
<td>indexBuildType</td>
<td>ConnectApi. CommerceSearch IndexBuildType</td>
<td>Build type of the index. Values are:</td>
<td>57.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Full</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Incremental</td>
<td></td>
</tr>
<tr>
<td>indexStatus</td>
<td>ConnectApi. CommerceSearch IndexStatus</td>
<td>Status of the index. Values are:</td>
<td>52.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Completed</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Failed</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• InProgress</td>
<td></td>
</tr>
<tr>
<td>indexUsage</td>
<td>ConnectApi. CommerceSearch IndexUsage</td>
<td>Usage of the index. Values are:</td>
<td>52.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Live</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• OutOfUse</td>
<td></td>
</tr>
<tr>
<td>isIncrementable</td>
<td>Boolean</td>
<td>Specifies whether the index allows incremental indexing (true) or not (false).</td>
<td>57.0</td>
</tr>
<tr>
<td>lastCatalogSnapshotTime</td>
<td>Datetime</td>
<td>Catalog snapshot time of the index.</td>
<td>57.0</td>
</tr>
<tr>
<td>message</td>
<td>String</td>
<td>Detailed message for the index status.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

ConnectApi.CommerceSearchIndexCollection

**ConnectApi.CommerceSearchIndexCollection**

Collection of indexes.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>indexes</td>
<td>List&lt;ConnectApi. CommerceSearch Index&gt;</td>
<td>List of up to two indexes. Returns the completed, live index and either the in-progress, out-of-use index or the most-recently-failed, out-of-use index.</td>
<td>52.0</td>
</tr>
</tbody>
</table>
### ConnectApi.CommerceSearchIndexLog

Search index log information.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>catalogSnapshotTime</td>
<td>Datetime</td>
<td>Catalog snapshot time of the index build.</td>
<td>57.0</td>
</tr>
<tr>
<td>completionDate</td>
<td>Datetime</td>
<td>Completion date of the index build.</td>
<td>57.0</td>
</tr>
<tr>
<td>createdById</td>
<td>String</td>
<td>ID of the user who initiated the index build.</td>
<td>57.0</td>
</tr>
<tr>
<td>indexBuildStatus</td>
<td>ConnectApi.CommerceSearchIndexStatus</td>
<td>Status of the index. Values are: - Completed - Failed - InProgress</td>
<td>57.0</td>
</tr>
<tr>
<td>indexBuildType</td>
<td>ConnectApi.CommerceSearchIndexBuildType</td>
<td>Build type of the index. Values are: - Full - Incremental</td>
<td>57.0</td>
</tr>
<tr>
<td>indexId</td>
<td>String</td>
<td>ID of the index build.</td>
<td>57.0</td>
</tr>
<tr>
<td>message</td>
<td>String</td>
<td>Detailed message for the index build status.</td>
<td>57.0</td>
</tr>
<tr>
<td>numberOfProducts</td>
<td>Integer</td>
<td>Number of new or changed products in the index build.</td>
<td>57.0</td>
</tr>
</tbody>
</table>

### ConnectApi.CommerceSearchIndexLogCollection

Collection of search index logs for a webstore.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>indexLogs</td>
<td>List&lt;ConnectApi.CommerceSearchIndexLog&gt;</td>
<td>List of up to 100 index logs sorted by most recent catalog snapshot time of the index.</td>
<td>57.0</td>
</tr>
</tbody>
</table>

### ConnectApi.Community

Experience Cloud site.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>allowChatter</td>
<td>Boolean</td>
<td>Specifies if guest users can access public groups without logging in.</td>
<td>31.0</td>
</tr>
<tr>
<td>AccessWithoutLogin</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>------------------------</td>
<td>----------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>allowMembersToFlag</td>
<td>Boolean</td>
<td>Specifies if members can flag content.</td>
<td>30.0</td>
</tr>
<tr>
<td>builderBasedSnaEnabled</td>
<td>Boolean</td>
<td>Specifies whether the Service Not Available page is an auto-generated Experience Builder-based page ( \text{true} ) or a static resource page that's set in Workspaces &gt; Administration &gt; Pages ( \text{false} ).</td>
<td>52.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Experience Builder URL for the site.</td>
<td>56.0</td>
</tr>
<tr>
<td>guestMemberVisibilityEnabled</td>
<td>Boolean</td>
<td>Specifies whether guest members can see other members ( \text{true} ) or not ( \text{false} ).</td>
<td>47.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>Site ID.</td>
<td>28.0</td>
</tr>
<tr>
<td>imageOptimizationCDNEnabled</td>
<td>Boolean</td>
<td>Specifies whether images are optimized for guest users on all devices for sites using Salesforce’s CDN for Digital Experiences ( \text{true} ) or not ( \text{false} ).</td>
<td>56.0</td>
</tr>
<tr>
<td>invitationsEnabled</td>
<td>Boolean</td>
<td>Specifies whether users can invite other external users.</td>
<td>28.0</td>
</tr>
<tr>
<td>knowledgeableEnabled</td>
<td>Boolean</td>
<td>Specifies whether knowledgeable people and endorsements are available for topics ( \text{true} ), or not ( \text{false} ).</td>
<td>30.0</td>
</tr>
<tr>
<td>loginUrl</td>
<td>String</td>
<td>Login URL for the site.</td>
<td>36.0</td>
</tr>
<tr>
<td>memberVisibilityEnabled</td>
<td>Boolean</td>
<td>Specifies whether members can see other members ( \text{true} ) or not ( \text{false} ).</td>
<td>45.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Site name.</td>
<td>28.0</td>
</tr>
<tr>
<td>nicknameDisplayEnabled</td>
<td>Boolean</td>
<td>Specifies whether nicknames are displayed.</td>
<td>32.0</td>
</tr>
<tr>
<td>privateMessagesEnabled</td>
<td>Boolean</td>
<td>Specifies whether members can send and receive private messages to and from other members ( \text{true} ) or not ( \text{false} ).</td>
<td>30.0</td>
</tr>
<tr>
<td>reputationEnabled</td>
<td>Boolean</td>
<td>Specifies whether reputation is calculated and displayed for members.</td>
<td>31.0</td>
</tr>
<tr>
<td>sendWelcomeEmail</td>
<td>Boolean</td>
<td>Specifies whether emails are sent to all new users when they join.</td>
<td>28.0</td>
</tr>
<tr>
<td>siteAsContainerEnabled</td>
<td>Boolean</td>
<td>Specifies whether the site is an Experience Builder site ( \text{true} ) or a Salesforce Tabs + Visualforce site ( \text{false} ).</td>
<td>41.0</td>
</tr>
<tr>
<td>siteUrl</td>
<td>String</td>
<td>URL for the site, which is your Experience Cloud sites domain plus a URL prefix. For example, MyDomainName.my.site.com/customers.</td>
<td>30.0</td>
</tr>
</tbody>
</table>

**Note:** If you're not using enhanced domains, your org’s Experience Cloud sites URL is different. For details, see My Domain URL Formats in Salesforce Help.
### ConnectApi.CommunityPage

Page of Experience Cloud sites.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>communities</td>
<td>List&lt;ConnectApi.Community&gt;</td>
<td>List of Experience Cloud sites the context user has access to.</td>
<td>28.0</td>
</tr>
<tr>
<td>total</td>
<td>Integer</td>
<td>Total number of Experience Cloud sites.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

### ConnectApi.CommunitySummary

Summary of an Experience Cloud site.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>18-character ID of the site.</td>
<td>41.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Localized name of the site.</td>
<td>41.0</td>
</tr>
</tbody>
</table>

SEE ALSO:  
ConnectApi.UserActivitySummary

### ConnectApi.CompanyVerifySummary

Company verify summary.

Subclass of ConnectApi.UserFeedEntityActivitySummary.
ConnectApi.ComplexSegment

Complex segments of field changes.
This class is abstract.
Subclass of ConnectApi.MessageSegment.
Superclass of ConnectApi.FieldChangeSegment.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>segments</td>
<td>List&lt;ConnectApi.MessageSegment&gt;</td>
<td>List of message segments.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

ConnectApi.CompoundRecordField

Record field that is a composite of subfields.
Subclass of ConnectApi.LabeledRecordField.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>fields</td>
<td>List&lt;ConnectApi.AbstractRecordField&gt;</td>
<td>Collection of subfields that make up the compound field.</td>
<td>29.0</td>
</tr>
</tbody>
</table>

ConnectApi.ConfirmHeldFOCapacityOutputRepresentation

Response to a request to confirm held fulfillment order capacity at one or more locations. Can correspond to one action call.
Subclass of ConnectApi.BaseOutputRepresentation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>confirmHeldFOCapacityResponses</td>
<td>List&lt;ConnectApi.ConfirmHeldFOCapacityResponseOutputRepresentation&gt;</td>
<td>List of responses to the requests to confirm held fulfillment order capacity at one or more locations.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

ConnectApi.ConfirmHeldFOCapacityResponseOutputRepresentation

Response to a request to confirm held fulfillment order capacity at one or more locations.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>capacityResponses</td>
<td>List&lt;ConnectApi.CapacityResponseOutputRepresentation&gt;</td>
<td>List of responses to the requests to confirm held fulfillment order capacity at individual locations.</td>
<td>55.0</td>
</tr>
</tbody>
</table>
## ConnectApi.Content

A file attached to a feed item.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>checksum</td>
<td>String</td>
<td>MD5 checksum for the file.</td>
<td>36.0</td>
</tr>
<tr>
<td>contentUrl</td>
<td>String</td>
<td>URL of the content for links.</td>
<td>36.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the attachment.</td>
<td>36.0</td>
</tr>
<tr>
<td>downloadUrl</td>
<td>String</td>
<td>URL to the content.</td>
<td>36.0</td>
</tr>
<tr>
<td>fileExtension</td>
<td>String</td>
<td>Extension of the file.</td>
<td>36.0</td>
</tr>
<tr>
<td>fileSize</td>
<td>String</td>
<td>Size of the file in bytes. If size can't be determined, returns unknown.</td>
<td>36.0</td>
</tr>
<tr>
<td>fileType</td>
<td>String</td>
<td>Type of file, such as PDF.</td>
<td>36.0</td>
</tr>
<tr>
<td>hasPdfPreview</td>
<td>Boolean</td>
<td>true if the file has a PDF preview available; false otherwise.</td>
<td>36.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>18-character ID of the content.</td>
<td>36.0</td>
</tr>
<tr>
<td>imageDetails</td>
<td>ConnectApi.ContentImageFileDetails</td>
<td>Image details, or null if the file isn't an image.</td>
<td>40.0</td>
</tr>
<tr>
<td>isInMyFileSync</td>
<td>Boolean</td>
<td>true if the file is synced with Salesforce Files Sync.</td>
<td>36.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> Salesforce Files Sync was retired on May 25, 2018.</td>
<td></td>
</tr>
<tr>
<td>mimeType</td>
<td>String</td>
<td>MIME type of the file.</td>
<td>36.0</td>
</tr>
<tr>
<td>renditionUrl</td>
<td>String</td>
<td>URL to the rendition resource for the file. For shared files, renditions process asynchronously after upload. For private files, renditions process when the first file preview is requested, and aren't available immediately after the file is uploaded.</td>
<td>36.0</td>
</tr>
<tr>
<td>renditionUrl 240By180</td>
<td>String</td>
<td>URL to the 240 x 180 pixel rendition resource for the file. For shared files, renditions process asynchronously after upload. For private files, renditions process when the first file preview is requested, and aren't available immediately after the file is uploaded.</td>
<td>36.0</td>
</tr>
<tr>
<td>renditionUrl 720By480</td>
<td>String</td>
<td>URL to the 720 x 480 pixel rendition resource for the file. For shared files, renditions process asynchronously after upload. For private files, renditions process when the first file preview is requested, and aren't available immediately after the file is uploaded.</td>
<td>36.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>---------------</td>
<td>---------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>sharingOption</td>
<td>ConnectApi.</td>
<td>Sharing option of the file. Values are:</td>
<td>36.0</td>
</tr>
<tr>
<td></td>
<td>FileSharingOption</td>
<td>• Allowed—Resharing of the file is allowed.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Restricted—Resharing of the file is restricted.</td>
<td></td>
</tr>
<tr>
<td>textPreview</td>
<td>String</td>
<td>Text preview of the file if available; null otherwise.</td>
<td>36.0</td>
</tr>
<tr>
<td>thumb120By90</td>
<td>String</td>
<td>Specifies the rendering status of the 120 x 90 preview image of the file. One of these values:</td>
<td>36.0</td>
</tr>
<tr>
<td>RenditionStatus</td>
<td></td>
<td>• Processing—Image is being rendered.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Failed—Rendering process failed.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Success—Rendering process was successful.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Na—Rendering is not available for this image.</td>
<td></td>
</tr>
<tr>
<td>thumb240By180</td>
<td>String</td>
<td>Specifies the rendering status of the 240 x 180 preview image of the file. One of these values:</td>
<td>36.0</td>
</tr>
<tr>
<td>RenditionStatus</td>
<td></td>
<td>• Processing—Image is being rendered.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Failed—Rendering process failed.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Success—Rendering process was successful.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Na—Rendering is not available for this image.</td>
<td></td>
</tr>
<tr>
<td>thumb720By480</td>
<td>String</td>
<td>Specifies the rendering status of the 720 x 480 preview image of the file. One of these values:</td>
<td>36.0</td>
</tr>
<tr>
<td>RenditionStatus</td>
<td></td>
<td>• Processing—Image is being rendered.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Failed—Rendering process failed.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Success—Rendering process was successful.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Na—Rendering is not available for this image.</td>
<td></td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>Title of the file.</td>
<td>36.0</td>
</tr>
<tr>
<td>versionId</td>
<td>String</td>
<td>Version ID of the file.</td>
<td>36.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.FilesCapability

**ConnectApi.ContentCapability**

If a comment has this capability, it has a file attachment.
Subclass of ConnectApi.FeedElementCapability.
For files attached to a feed post (instead of a comment) in version 36.0 and later, use ConnectApi.FilesCapability.
If content is deleted from a feed element after it's posted or if the access to the content is changed to private, the ConnectApi.ContentCapability exists, however most of its properties are null.
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>checksum</td>
<td>String</td>
<td>MD5 checksum for the file.</td>
<td>32.0</td>
</tr>
<tr>
<td>contentUrl</td>
<td>String</td>
<td>URL of the content for links and Google docs.</td>
<td>32.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the attachment.</td>
<td>32.0</td>
</tr>
<tr>
<td>downloadUrl</td>
<td>String</td>
<td>URL to the content.</td>
<td>32.0</td>
</tr>
<tr>
<td>fileExtension</td>
<td>String</td>
<td>Extension of the file.</td>
<td>32.0</td>
</tr>
<tr>
<td>fileSize</td>
<td>String</td>
<td>Size of the file in bytes. If size cannot be determined, returns Unknown.</td>
<td>32.0</td>
</tr>
<tr>
<td>fileType</td>
<td>String</td>
<td>Type of file.</td>
<td>32.0</td>
</tr>
<tr>
<td>hasPdfPreview</td>
<td>Boolean</td>
<td>true if the file has a PDF preview available, false otherwise.</td>
<td>32.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>18-character ID of the content.</td>
<td>32.0</td>
</tr>
<tr>
<td>isInMyFileSync</td>
<td>Boolean</td>
<td>true if the file is synced with Salesforce Files Sync; false otherwise.</td>
<td>32.0</td>
</tr>
<tr>
<td>mimeType</td>
<td>String</td>
<td>MIME type of the file.</td>
<td>32.0</td>
</tr>
<tr>
<td>renditionUrl</td>
<td>String</td>
<td>URL to the rendition resource for the file. Renditions are processed asynchronously and may not be available immediately after the file has been uploaded.</td>
<td>32.0</td>
</tr>
<tr>
<td>renditionUrl240By180</td>
<td>String</td>
<td>URL to the 240x180 size rendition resource for the file. Renditions are processed asynchronously and may not be available immediately after the file has been uploaded.</td>
<td>32.0</td>
</tr>
<tr>
<td>renditionUrl720By480</td>
<td>String</td>
<td>URL to the 720x480 size rendition resource for the file. Renditions are processed asynchronously and may not be available immediately after the file has been uploaded.</td>
<td>32.0</td>
</tr>
</tbody>
</table>
| sharingOption | ConnectApi.FileSharingOption | Sharing option of the file. Values are:  
  • Allowed—Resharing of the file is allowed.  
  • Restricted—Resharing of the file is restricted. | 35.0 |
| textPreview   | String | Text preview of the file if available, null otherwise. The maximum number of characters is 200. | 32.0 |

*Note: Salesforce Files Sync was retired on May 25, 2018.*
### Available Version

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>thumb120By90RenditionStatus</td>
<td>String</td>
<td>The status of the rendering of the 120x90 pixel sized preview image of the file. Should be either Processing, Failed, Success, or Na if unavailable.</td>
<td>32.0</td>
</tr>
<tr>
<td>thumb240By180RenditionStatus</td>
<td>String</td>
<td>The status of the rendering of the 240x180 pixel sized preview image of the file. Should be either Processing, Failed, Success, or Na if unavailable.</td>
<td>32.0</td>
</tr>
<tr>
<td>thumb720By480RenditionStatus</td>
<td>String</td>
<td>The status of the rendering of the 720x480 pixel sized preview image of the file. Should be either Processing, Failed, Success, or Na if unavailable.</td>
<td>32.0</td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>Title of the file.</td>
<td>32.0</td>
</tr>
<tr>
<td>versionId</td>
<td>String</td>
<td>Version ID of the file.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- [ConnectApi.CommentCapabilities](#)

### ConnectApi.ContentHubAllowedItemTypeCollection

The item types that the context user is allowed to create in a repository folder.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>allowedItemTypes</td>
<td>List&lt;ConnectApi.ContentHubItemTypeSummary&gt;</td>
<td>A collection of item types that the context user is allowed to create in a repository folder.</td>
<td>39.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ContentHubFieldDefinition

A field definition.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>displayName</td>
<td>String</td>
<td>Label or caption of this field.</td>
<td>39.0</td>
</tr>
<tr>
<td>isMandatory</td>
<td>Boolean</td>
<td>Specifies whether this field is mandatory for the item type.</td>
<td>39.0</td>
</tr>
<tr>
<td>maxLength</td>
<td>Integer</td>
<td>Maximum length of the value of this field.</td>
<td>39.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the field.</td>
<td>39.0</td>
</tr>
</tbody>
</table>
### ConnectApi.ContentHubItemTypeDetail

The details of an item type associated with a repository folder.
Subclass of ConnectApi.AbstractContentHubItemType.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>fields</td>
<td>List&lt;ConnectApi.ContentHubFieldDefinition&gt;</td>
<td>A list of fields that the context user is allowed to set in the metadata of this item type.</td>
<td>39.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ContentHubItemTypeSummary

The summary of an item type associated with a repository folder.
Subclass of ConnectApi.AbstractContentHubItemType.
No additional properties.

SEE ALSO:
ConnectApi.ContentHubAllowedItemTypeCollection

### ConnectApi.ContentHubPermissionType

A permission type.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>Internal ID of the permission type in the repository.</td>
<td>39.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.ContentHubItemTypeDetail
### Available Version Description Type Property Name

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>label</td>
<td>String</td>
<td>Label as returned by the repository.</td>
<td>39.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

- `ConnectApi.ExternalFilePermissionInformation`

### ConnectApi.ContentHubProviderType

The type of repository.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>label</td>
<td>String</td>
<td>Localized label of the provider type.</td>
<td>39.0</td>
</tr>
<tr>
<td>type</td>
<td>String</td>
<td>Provider type. One of these values:</td>
<td>39.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ContentHubBox</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ContentHubGDrive</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ContentHubSharepoint</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ContentHubSharepointOffice365</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ContentHubSharepointOneDrive</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• SimpleUrl</td>
<td></td>
</tr>
</tbody>
</table>

SEE ALSO:

- `ConnectApi.ContentHubRepository`

### ConnectApi.ContentHubRepository

A repository.

Subclass of `ConnectApi.ActorWithId`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>authentication</td>
<td><code>ConnectApi.ContentHubRepositoryAuthentication</code></td>
<td>Repository authentication information.</td>
<td>40.0</td>
</tr>
<tr>
<td>features</td>
<td><code>ConnectApi.ContentHubRepositoryFeatures</code></td>
<td>Repository features.</td>
<td>39.0</td>
</tr>
<tr>
<td>label</td>
<td>String</td>
<td>Repository label.</td>
<td>39.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Repository name.</td>
<td>39.0</td>
</tr>
<tr>
<td>providerType</td>
<td><code>ConnectApi.ContentHubProviderType</code></td>
<td>Repository provider type.</td>
<td>39.0</td>
</tr>
</tbody>
</table>
**rootFolderItemsUrl**  
*String*  
URL to the list of items in the repository root folder.  
39.0

**SEE ALSO:**  
ConnectApi.ContentHubRepositoryCollection

### ConnectApi.ContentHubRepositoryAuthentication

Authentication information for a repository.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
</table>
| authFlowUrl         | String              | Depends on the authProtocol.  
• NoAuthentication—null.  
• Oauth—URL to start the OAuth flow.  
• Password—URL to the authentication settings for external systems. | 40.0              |

| authProtocol        | ConnectApi.         | Authentication protocol used for the repository.  
Values are:  
• NoAuthentication—Repository doesn’t require authentication.  
• Oauth—Repository uses OAuth authentication protocol.  
• Password—Repository uses user name and password authentication protocol. | 40.0              |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>userHasAuthSettings</td>
<td>Boolean</td>
<td>Specifies whether the user has credentials or the administrator configured the external data source to use the same set of credentials for every user (true). Otherwise, false.</td>
<td>40.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**  
ConnectApi.ContentHubRepository

### ConnectApi.ContentHubRepositoryCollection

A collection of repositories.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>URL to the current page of repositories.</td>
<td>39.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>URL to the next page of repositories, or null if there isn’t a next page.</td>
<td>39.0</td>
</tr>
</tbody>
</table>
### ConnectApi.ContentHubRepositoryFeatures

The features of a repository.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>canBrowse</td>
<td>Boolean</td>
<td>Specifies whether the repository’s folder hierarchy can be browsed (true) or not (false).</td>
<td>39.0</td>
</tr>
<tr>
<td>canSearch</td>
<td>Boolean</td>
<td>Specifies whether the repository can be searched (true) or not (false).</td>
<td>39.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.ContentHubRepository

### ConnectApi.ContentImageFileDetails

Image file details.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>height</td>
<td>Integer</td>
<td>Image’s height in pixels.</td>
<td>40.0</td>
</tr>
<tr>
<td>imageFormat</td>
<td>String</td>
<td>Image’s format.</td>
<td>40.0</td>
</tr>
<tr>
<td>orientation</td>
<td>String</td>
<td>Image’s EXIF orientation value, if present.</td>
<td>40.0</td>
</tr>
<tr>
<td>width</td>
<td>Integer</td>
<td>Image’s width in pixels.</td>
<td>40.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.InlineImageSegment

### ConnectApi.ContractOutputRepresentation

Contract list.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>List&lt;String&gt;</td>
<td>Record IDs of the contacts.</td>
<td>56.0</td>
</tr>
</tbody>
</table>
ConnectApi.ConversationApplicationDefinitionDetailRepresentation

Information about the conversation application definition.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>botInfo</td>
<td>ConnectApi.BotInfoRepresenta</td>
<td>Basic information of the bot associated with this conversation application.</td>
<td>54.0</td>
</tr>
<tr>
<td>errorMessage</td>
<td>String</td>
<td>Error message for the failed get operation.</td>
<td>54.0</td>
</tr>
<tr>
<td>integration</td>
<td>ConnectApi.ConversationAppli</td>
<td>Conversation application integration types. Values are:</td>
<td>54.0</td>
</tr>
<tr>
<td>integrationName</td>
<td>String</td>
<td>Name of the conversation application.</td>
<td>54.0</td>
</tr>
<tr>
<td>isSuccess</td>
<td>Boolean</td>
<td>Success indicator of the get operation.</td>
<td>54.0</td>
</tr>
<tr>
<td>runtimeUrl</td>
<td>String</td>
<td>Base URL of the bot runtime API.</td>
<td>54.0</td>
</tr>
</tbody>
</table>

ConnectApi.CouponCodeRedemptionResult

Coupon code redemption result.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>availableRedemptions</td>
<td>Integer</td>
<td>Number of coupon code redemptions available.</td>
<td>58.0</td>
</tr>
<tr>
<td>couponCode</td>
<td>String</td>
<td>Coupon code.</td>
<td>58.0</td>
</tr>
<tr>
<td>errorMsg</td>
<td>String</td>
<td>Error message when coupon code redemption isn’t successful.</td>
<td>58.0</td>
</tr>
<tr>
<td>isSuccess</td>
<td>Boolean</td>
<td>Specifies whether increasing or decreasing the coupon code redemption is successful (true) or not (false).</td>
<td>58.0</td>
</tr>
<tr>
<td>redemptionLimit</td>
<td>Integer</td>
<td>Number of coupon code redemptions allowed.</td>
<td>58.0</td>
</tr>
</tbody>
</table>

ConnectApi.CouponCodeRedemptionCollection

Collection of coupon code redemption results.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>couponCode</td>
<td>List&lt;ConnectApi.CouponCodeRedemptionResult&gt;</td>
<td>List of coupon code redemption results.</td>
<td>58.0</td>
</tr>
</tbody>
</table>
**ConnectApi.CreateCreditMemoOutputRepresentation**

ID of a created Credit Memo.

Subclass of `ConnectApi.BaseOutputRepresentation`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>creditMemoId</td>
<td>String</td>
<td>ID of the created Credit Memo.</td>
<td>48.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ChangeOrdersInvoiceOutputRepresentation**

List of IDs of invoices created for change orders.

Subclass of `ConnectApi.BaseInvoiceOutputRepresentation`.

No additional properties.

SEE ALSO:

- `createMultipleInvoices(invoicesInput)`
- `ConnectApi.CreateMultipleInvoicesFromChangeOrdersOutputRepresentation`

**ConnectApi.CreateMultipleInvoicesFromChangeOrdersOutputRepresentation**

List of lists of invoices created from change orders for fees.

Subclass of `ConnectApi.BaseOutputRepresentation`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>invoices</td>
<td>List&lt;<code>ConnectApi.ChangeOrdersInvoiceOutputRepresentation</code>&gt;</td>
<td>List of IDs of invoices created from change orders for fees. Include these invoice IDs when calling Ensure Refunds for the return that the fees applied to.</td>
<td>56.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

- `createMultipleInvoices(invoicesInput)`

**ConnectApi.CreateOrderPaymentSummaryOutputRepresentation**

ID of the created Order Payment Summary.

Subclass of `ConnectApi.BaseOutputRepresentation`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>orderPaymentSummaryId</td>
<td>String</td>
<td>ID of the Order Payment Summary.</td>
<td>48.0</td>
</tr>
</tbody>
</table>
# ConnectApi.Credential

**Credential.**

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>authentication Protocol</td>
<td>ConnectApi. CredentialAuthentication Protocol</td>
<td>Authentication protocol of the external credential. Values are:</td>
<td>56.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsSv4</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Custom</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Jwt</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• OAuth</td>
<td></td>
</tr>
<tr>
<td>authentication ProtocolVariant</td>
<td>ConnectApi. CredentialAuthentication ProtocolVariant</td>
<td>Authentication protocol variant of the external credential. Values are:</td>
<td>57.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AwsSv4_STS—AWS Signature Version 4 with Security Token Service.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ClientCredentialsClientSecret—OAuth 2.0 Client Credentials client secret.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ClientCredentialsJwtAssertion—OAuth 2.0 Client Credentials JSON Web Token assertion.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• JwtBearer—OAuth 2.0 JSON Web Token bearer flow.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NoAuthentication—No authentication.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• RolesAnywhere—AWS Signature Version 4 with Identity and Access Management (IAM) Roles Anywhere.</td>
<td></td>
</tr>
<tr>
<td>authentication Status</td>
<td>ConnectApi. CredentialAuthentication Status</td>
<td>Status of the credential authentication. Values are:</td>
<td>56.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Configured—Credential has all required credentials for at least one principal.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NotConfigured—Credential isn’t configured.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Unknown—Credential status can’t be determined because the authentication protocol is custom.</td>
<td></td>
</tr>
<tr>
<td>credentials</td>
<td>Map&lt;String, ConnectApi. CredentialValue&gt;</td>
<td>Map of protocol-specific credentials.</td>
<td>56.0</td>
</tr>
<tr>
<td>externalCredential</td>
<td>String</td>
<td>Fully qualified developer name of the external credential.</td>
<td>56.0</td>
</tr>
<tr>
<td>principalName</td>
<td>String</td>
<td>Name of the external credential named principal.</td>
<td>56.0</td>
</tr>
</tbody>
</table>
### ConnectApi.CredentialPrincipal

**Property Name** | **Type** | **Description** | **Available Version**
--- | --- | --- | ---
principalType | ConnectApi.CredentialPrincipal Type | Type of credential principal. Values are: 
- AwsStsPrincipal 
- NamedPrincipal 
- PerUserPrincipal | 56.0

### ConnectApi.CredentialCustomHeader

Credential custom header.

**Property Name** | **Type** | **Description** | **Available Version**
--- | --- | --- | ---
headerName | String | Header name. | 57.0
headerValue | String | Header value that can contain formulas. | 57.0
id | String | ID of the customer header parameter. | 58.0
sequenceNumber | Integer | Sequence number of the header. The sequence number determines the order of the header. | 57.0

**SEE ALSO:**
- ConnectApi.ExternalCredential
- ConnectApi.NamedCredential

### ConnectApi.CredentialValue

Credential value.

**Property Name** | **Type** | **Description** | **Available Version**
--- | --- | --- | ---
encrypted | Boolean | Specifies whether the credential value is encrypted (true) or not (false). | 56.0
revision | Integer | Revision number of a short-lived credential. | 58.0
value | String | Value of the credential. | 56.0

**SEE ALSO:**
- ConnectApi.Credential

### ConnectApi.CurrencyRecordField

Record field containing a currency value.

Subclass of ConnectApi.LabeledRecordField.
**ConnectApi.CustomListAudienceCriteria**

Criteria for the custom list type of custom recommendation audience.
Subclass of `ConnectApi.AudienceCriteria`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>memberCount</td>
<td>Integer</td>
<td>Total number of members in the custom recommendation audience.</td>
<td>36.0</td>
</tr>
<tr>
<td>members</td>
<td><code>ConnectApi.UserReferencePage</code></td>
<td>Members of the custom recommendation audience.</td>
<td>36.0</td>
</tr>
</tbody>
</table>

**ConnectApi.DashboardComponentSnapshot**

Represents both dashboard component snapshots and alerts you receive when a dashboard component value crosses a threshold.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>componentId</td>
<td>String</td>
<td>18-character ID of the dashboard component.</td>
<td>32.0</td>
</tr>
<tr>
<td>componentName</td>
<td>String</td>
<td>The dashboard component name.</td>
<td>32.0</td>
</tr>
<tr>
<td>dashboardBodyText</td>
<td>String</td>
<td>Display this text next to the actor in the feed element. Use this text in place of the default body text.</td>
<td>32.0</td>
</tr>
<tr>
<td>dashboardId</td>
<td>String</td>
<td>18-character ID of the dashboard.</td>
<td>32.0</td>
</tr>
<tr>
<td>dashboardName</td>
<td>String</td>
<td>The name of the dashboard.</td>
<td>32.0</td>
</tr>
<tr>
<td>fullSizeImageUrl</td>
<td>String</td>
<td>The source URL to retrieve the full-size image of a snapshot. Access this URL with OAuth credentials.</td>
<td>32.0</td>
</tr>
<tr>
<td>lastRefreshDate</td>
<td>Datetime</td>
<td>ISO 8601 date specifying when this dashboard component was last refreshed.</td>
<td>32.0</td>
</tr>
<tr>
<td>lastRefreshDateDisplayText</td>
<td>String</td>
<td>Display text for the last refresh date, for example, “Last Refreshed on October 31, 2013.”</td>
<td>32.0</td>
</tr>
<tr>
<td>runningUser</td>
<td><code>ConnectApi.UserSummary</code></td>
<td>The running user of the dashboard at the time the snapshot was posted. This value may be <code>null</code>. Each dashboard has a running user, whose security settings determine which data to display in a dashboard.</td>
<td>32.0</td>
</tr>
<tr>
<td>thumbnailUrl</td>
<td>String</td>
<td>The source URL to retrieve the thumbnail image of a snapshot. Access this URL with OAuth credentials.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- `ConnectApi.DashboardComponentSnapshotCapability`
- `ConnectApi.DatacloudCompanies`
ConnectApi.DashboardComponentSnapshotCapability

If a feed element has this capability, it has a dashboard component snapshot. A snapshot is a static image of a dashboard component at a specific point in time.

Subclass of ConnectApi.FeedElementCapability.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>dashboardComponent Snapshot</td>
<td>ConnectApi.DashboardComponentSnapshot</td>
<td>The dashboard component snapshot.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

ConnectApi.FeedElementCapabilities

ConnectApi.DateRecordField

Record field containing a date.

Subclass of ConnectApi.LabeledRecordField.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>dateValue</td>
<td>Datetime</td>
<td>Date that a machine can read. Ignore the trailing 00:00:00.000Z characters.</td>
<td>29.0</td>
</tr>
</tbody>
</table>

ConnectApi.DeleteIntent

Delete intent for a social post.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>managedSocialAccount</td>
<td>ConnectApi. ManagedSocialAccount</td>
<td>Managed social account that deletes the social post.</td>
<td>45.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

ConnectApi.DeleteIntents

ConnectApi.DeleteIntents

List of delete intents for a social post.
### ConnectApi.DeleteSocialPostIntent
Delete intent for the social post.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>socialAccountId</td>
<td>String</td>
<td>ID of the social account that deletes the social post.</td>
<td>46.0</td>
</tr>
<tr>
<td>socialPostId</td>
<td>String</td>
<td>ID of the social post to delete.</td>
<td>46.0</td>
</tr>
</tbody>
</table>

### ConnectApi.DigestJob
Represents a successfully enqueued API digest job request.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
</table>
| period | ConnectApi.DigestPeriod | Time period that's included in a Chatter email digest. Values are:  
• DailyDigest—The email includes up to the 50 latest posts from the previous day.  
• WeeklyDigest—The email includes up to the 50 latest posts from the previous week. | 37.0 |

### ConnectApi.DirectMessageCapability
If a feed element has this capability, it’s a direct message.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>memberChanges</td>
<td>ConnectApi.DirectMessageMemberActivityPage</td>
<td>Member activities of the direct message, with the most recent activity first.</td>
<td>40.0</td>
</tr>
<tr>
<td>members</td>
<td>ConnectApi.DirectMessageMemberPage</td>
<td>Members included in the direct message.</td>
<td>39.0</td>
</tr>
<tr>
<td>originalMembers</td>
<td>ConnectApi.DirectMessageMemberPage</td>
<td>Original members of the direct message.</td>
<td>40.0</td>
</tr>
</tbody>
</table>
### ConnectApi.DirectMessageMemberActivity

Direct message member activity.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>activityDate</td>
<td>Datetime</td>
<td>Direct message member activity date.</td>
<td>40.0</td>
</tr>
<tr>
<td>actor</td>
<td>ConnectApi.UserSummary</td>
<td>User who changed the direct message membership.</td>
<td>40.0</td>
</tr>
<tr>
<td>membersAdded</td>
<td>ConnectApi.DirectMessageMemberPage</td>
<td>Members added to the direct message as part of the activity.</td>
<td>40.0</td>
</tr>
<tr>
<td>membersRemoved</td>
<td>ConnectApi.DirectMessageMemberPage</td>
<td>Members removed from the direct message as part of the activity.</td>
<td>40.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.DirectMessageMemberActivityPage

### ConnectApi.DirectMessageMemberActivityPage

A page of direct message member activities.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>activities</td>
<td>List&lt;ConnectApi.DirectMessageMemberActivity&gt;</td>
<td>Collection of direct message member activities.</td>
<td>40.0</td>
</tr>
<tr>
<td>currentPageToken</td>
<td>String</td>
<td>Token identifying the current page.</td>
<td>40.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>40.0</td>
</tr>
<tr>
<td>nextPageToken</td>
<td>String</td>
<td>Token identifying the next page, or null if there isn't a next page.</td>
<td>40.0</td>
</tr>
</tbody>
</table>
### ConnectApi.DirectMessageMemberPage

A collection of direct message members.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageToken</td>
<td>String</td>
<td>Page token to access the current page of direct message members.</td>
<td>39.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>URL to the current page of direct message members.</td>
<td>39.0</td>
</tr>
<tr>
<td>nextPageToken</td>
<td>String</td>
<td>Page token to access the next page of direct message members.</td>
<td>39.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>URL to the next page of direct message members.</td>
<td>39.0</td>
</tr>
<tr>
<td>users</td>
<td>List&lt;ConnectApi.UserSummary&gt;</td>
<td>Collection of direct message members.</td>
<td>39.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.DirectMessageCapability
- ConnectApi.DirectMessageCapability
- ConnectApi.DirectMessageMemberActivity

### ConnectApi.DistanceCalculationOutputRepresentation

Shipping distance data for a set of inventory locations.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>averageDistance</td>
<td>Double</td>
<td>The average distance from the locations to the order recipient.</td>
<td>51.0</td>
</tr>
<tr>
<td>locations</td>
<td>List&lt;ConnectApi.LocationOutputRepresentation&gt;</td>
<td>The list of locations and their distances to the order recipient.</td>
<td>51.0</td>
</tr>
<tr>
<td>rank</td>
<td>Integer</td>
<td>This result’s rank among all results by average distance to the order recipient.</td>
<td>51.0</td>
</tr>
</tbody>
</table>
ConnectApi.DistinctFacetValue

Distinct facet value.
This class is a subclass of ConnectApi.FacetValue.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>displayName</td>
<td>String</td>
<td>Display name of the facet value.</td>
<td>52.0</td>
</tr>
<tr>
<td>nameOrId</td>
<td>String</td>
<td>Developer name of the attribute.</td>
<td>52.0</td>
</tr>
<tr>
<td>productCount</td>
<td>Long</td>
<td>Number of products in the search result that match the facet value.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.DistinctValueSearchFacet

ConnectApi.DistinctValueSearchFacet

Facet with distinct values in product search results.
This class is a subclass of ConnectApi.SearchFacet.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>values</td>
<td>List&lt;ConnectApi.DistinctFacetValue&gt;</td>
<td>Values of the facet found in the search result. Sorted by display name in alphabetical order.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

ConnectApi.DownVoteSummary

Summary of a downvote.
Subclass of ConnectApi.UserFeedEntityActivitySummary.
No additional properties.

ConnectApi.EditCapability

If a feed element or comment has this capability, users who have permission can edit it.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isEditRestricted</td>
<td>Boolean</td>
<td>Specifies whether editing this feed element or comment is restricted. If true, the context user can’t edit this feed element or comment. If false, the context user may or may not have permission to edit this feed element or comment. To determine if the context user can edit a feed element or comment, use the isFeedElementEditableByMe(communityId, feedElementId) or</td>
<td>34.0</td>
</tr>
</tbody>
</table>
### isCommentEditableByMe(communityId, commentId) method.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isEditableByMeUrl</td>
<td>String</td>
<td>The URL to check if the context user is able to edit this feed element or comment.</td>
<td>34.0</td>
</tr>
<tr>
<td>lastEditedBy</td>
<td>ConnectApi.Actor</td>
<td>Who last edited this feed element or comment.</td>
<td>34.0</td>
</tr>
<tr>
<td>lastEditedDate</td>
<td>Datetime</td>
<td>The most recent edit date of this feed element or comment.</td>
<td>34.0</td>
</tr>
<tr>
<td>latestRevision</td>
<td>Integer</td>
<td>The most recent revision of this feed element or comment.</td>
<td>34.0</td>
</tr>
<tr>
<td>relativeLastEditedDate</td>
<td>String</td>
<td>Relative last edited date, for example, “2h ago.”</td>
<td>34.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.CommentCapabilities
- ConnectApi.FeedElementCapabilities

## ConnectApi.EmailAddress

Email address.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>displayName</td>
<td>String</td>
<td>The display name for the email address.</td>
<td>29.0</td>
</tr>
<tr>
<td>emailAddress</td>
<td>String</td>
<td>The email address.</td>
<td>29.0</td>
</tr>
<tr>
<td>relatedRecord</td>
<td>ConnectApi.RecordSummary</td>
<td>The summary of a related record, for example, a contact or user summary.</td>
<td>36.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.EmailMessageCapability

## ConnectApi.EmailAttachment

An email attachment in an email message.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>attachment</td>
<td>ConnectApi.RecordSummary</td>
<td>Record summary of the attachment.</td>
<td>36.0</td>
</tr>
<tr>
<td>contentType</td>
<td>String</td>
<td>Type of attachment.</td>
<td>36.0</td>
</tr>
</tbody>
</table>
## ConnectApi.EmailMessageCapability

If a feed element has this capability, it has an email message from a case.

Subclass of **ConnectApi.FeedElementCapability**.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>attachments</td>
<td>List&lt;ConnectApi.EmailAttachment&gt;</td>
<td>Attachments in the email message.</td>
<td>36.0</td>
</tr>
<tr>
<td>bccAddresses</td>
<td>List&lt;ConnectApi.EmailAddress&gt;</td>
<td>BCC addresses for the email message.</td>
<td>36.0</td>
</tr>
<tr>
<td>body</td>
<td>String</td>
<td>Body of the email message.</td>
<td>36.0</td>
</tr>
<tr>
<td>ccAddresses</td>
<td>List&lt;ConnectApi.EmailAddress&gt;</td>
<td>CC addresses for the email message.</td>
<td>36.0</td>
</tr>
</tbody>
</table>
| direction     | ConnectApi.EmailMessageDirection | Direction of the email message. Values are:
|               |      | - Inbound—An inbound message (sent by a customer). | 32.0 |

### SEE ALSO:

**ConnectApi.EmailMergeFieldCollectionInfo**

The merge fields for an object.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>mergeFields</td>
<td>List&lt;String&gt;</td>
<td>List of merge fields for a single object.</td>
<td>39.0</td>
</tr>
</tbody>
</table>

### SEE ALSO:

**ConnectApi.EmailMergeFieldInfo**

The map for objects and their merge fields.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>entityToMergeFieldsMap</td>
<td>Map&lt;String, ConnectApi.EmailMergeFieldCollectionInfo&gt;</td>
<td>Map for multiple objects and their merge field collections.</td>
<td>39.0</td>
</tr>
</tbody>
</table>
## Available Version

### Description

- **Outbound**—An outbound message (sent to a customer by a support agent).

### Type

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>emailMessageId</td>
<td>String</td>
<td>ID of the email message.</td>
<td>32.0</td>
</tr>
<tr>
<td>fromAddress</td>
<td>ConnectApi.EmailAddress</td>
<td>From address for the email message.</td>
<td>36.0</td>
</tr>
<tr>
<td>htmlExpand</td>
<td>Integer</td>
<td>Start location of previous email thread.</td>
<td>47.0</td>
</tr>
<tr>
<td>isRichText</td>
<td>Boolean</td>
<td>Indicates whether the body of the email message is in rich text format.</td>
<td>36.0</td>
</tr>
<tr>
<td>status</td>
<td>ConnectApi.EmailMessageStatus</td>
<td>Status of an email message on a case. Values are:</td>
<td>47.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- DraftStatus</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- ForwardedStatus</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- NewStatus</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- ReadStatus</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- RepliedStatus</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- SentStatus</td>
<td></td>
</tr>
<tr>
<td>subject</td>
<td>String</td>
<td>Subject of the email message.</td>
<td>32.0</td>
</tr>
<tr>
<td>textBody</td>
<td>String</td>
<td>Body of the email message.</td>
<td>32.0–35.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Important:</strong> In version 36.0 and later, use the body property.</td>
<td></td>
</tr>
<tr>
<td>toAddresses</td>
<td>List&lt;ConnectApi.EmailAddress&gt;</td>
<td>To addresses of the email message.</td>
<td>32.0</td>
</tr>
<tr>
<td>totalAttachments</td>
<td>Integer</td>
<td>Total number of attachments in the email message.</td>
<td>38.0</td>
</tr>
</tbody>
</table>

### SEE ALSO:

- ConnectApi.FeedElementCapabilities

## ConnectApi.Emoji

An emoji.

### Property Name

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>category</td>
<td>String</td>
<td>Emoji category.</td>
<td>39.0</td>
</tr>
<tr>
<td>shortcut</td>
<td>String</td>
<td>Emoji shortcut.</td>
<td>39.0</td>
</tr>
</tbody>
</table>
**ConnectApi.EmojiCollection**

A collection of emoji.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>emojis</td>
<td>List&lt;ConnectApi.Emoji&gt;</td>
<td>A collection of emoji.</td>
<td>39.0</td>
</tr>
</tbody>
</table>

**ConnectApi.EnhancedLinkCapability**

If a feed element has this capability, it has a link that may contain supplemental information like an icon, a title, and a description.

Subclass of ConnectApi.FeedElementCapability.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>String</td>
<td>A description with a 500 character limit.</td>
<td>32.0</td>
</tr>
<tr>
<td>icon</td>
<td>ConnectApi.Icon</td>
<td>A icon.</td>
<td>32.0</td>
</tr>
<tr>
<td>linkRecordId</td>
<td>String</td>
<td>A ID associated with the link if the link URL refers to a Salesforce record.</td>
<td>32.0</td>
</tr>
<tr>
<td>linkUrl</td>
<td>String</td>
<td>A link URL to a detail page if available content can't display inline.</td>
<td>32.0</td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>A title to a detail page.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

**ConnectApi.EnsureFundsAsyncOutputRepresentation**

ID of the asynchronous background operation. This output only includes the operation ID, regardless of whether a call is made to an external payment gateway. It doesn't include any errors from the operation.

Subclass of ConnectApi.BaseAsyncOutputRepresentation.
No additional properties.

SEE ALSO:
ensureFundsAsync(orderSummaryId, ensureFundsInput)

ConnectApi.EnsureRefundsAsyncOutputRepresentation
ID of the asynchronous background operation. This output only includes the operation ID, regardless of whether a call is made to an external payment gateway. It doesn’t include any errors from the operation.
Subclass of ConnectApi.BaseAsyncOutputRepresentation.
No additional properties.

SEE ALSO:
ensureRefundsAsync(orderSummaryId, ensureRefundsInput)

ConnectApi.EntityLabel
An entity’s label.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>label</td>
<td>String</td>
<td>Localized singular label of the entity.</td>
<td>40.0</td>
</tr>
<tr>
<td>labelPlural</td>
<td>String</td>
<td>Localized plural label of the entity.</td>
<td>40.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.RecordSummary

ConnectApi.EntityLinkSegment
Entity link segment.
Subclass of ConnectApi.MessageSegment.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>motif</td>
<td>ConnectApi.Motif Class</td>
<td>A set of small, medium, and large icons that indicate whether the entity is a file, group, record, or user. The motif can also contain the object’s base color.</td>
<td>28.0</td>
</tr>
<tr>
<td>reference</td>
<td>ConnectApi.Reference</td>
<td>A reference to the link object if applicable, otherwise, null.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

ConnectApi.EntityRecommendation
A Chatter, custom, or static recommendation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>actOnUrl</td>
<td>String</td>
<td>For user, file, group, topic, and record entity types, use this Connect REST URL with a POST request to take action on the recommendation. For <code>ConnectApi.RecommendedObject</code> entity types, such as custom recommendations, use the <code>actionUrl</code> property of the <code>ConnectApi.PlatformAction</code> to take action on the recommendation.</td>
<td>32.0</td>
</tr>
</tbody>
</table>
| action         | `ConnectApi.RecommendationActionType` | Specifies the action to take on a recommendation.  
- follow—Follow a file, record, topic, or user.  
- join—Join a group.  
- view—View a file, group, article, record, user, custom, or static recommendation. | 32.0              |
| entity         | `ConnectApi.Actor`  | The entity with which the receiver is recommended to take action.                                                                                                                                                                                                                                                                         | 32.0              |

**ConnectApi.ErrorResponse**

Base error response.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>errorCode</td>
<td>String</td>
<td>Error code.</td>
<td>48.0</td>
</tr>
<tr>
<td>message</td>
<td>String</td>
<td>More error detail, if available.</td>
<td>48.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

- `ConnectApi.BaseOutputRepresentation`  
- `ConnectApi.Extension`  

**ConnectApi.Extension**

An extension.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>alternativeRepresentation</td>
<td><code>ConnectApi.Alternative</code></td>
<td>Alternative representation of the extension.</td>
<td>40.0</td>
</tr>
<tr>
<td>attachmentId</td>
<td>String</td>
<td>Attachment ID of the extension.</td>
<td>41.0</td>
</tr>
<tr>
<td>extensionId</td>
<td>String</td>
<td>ID of the extension.</td>
<td>40.0</td>
</tr>
<tr>
<td>payload</td>
<td>String</td>
<td>Payload associated with the extension.</td>
<td>40.0</td>
</tr>
</tbody>
</table>
### payloadVersion
- **Type:** String
- **Description:** Payload version that identifies the structure of the payload associated with the extension.
- **Available Version:** 40.0

**SEE ALSO:**
- ConnectApi.ExtensionsCapability

### ConnectApi.ExtensionDefinition

An extension’s definition.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>canAccess</td>
<td>Boolean</td>
<td>Indicates whether users can access the extension when it’s associated with a feed element.</td>
<td>40.0</td>
</tr>
<tr>
<td>canCreate</td>
<td>Boolean</td>
<td>Indicates whether users can create a feed element with the extension in the org.</td>
<td>40.0</td>
</tr>
<tr>
<td>createdDate</td>
<td>Datetime</td>
<td>Date when the extension was created.</td>
<td>40.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the extension.</td>
<td>40.0</td>
</tr>
<tr>
<td>iconUrl</td>
<td>String</td>
<td>URL to the icon for the extension.</td>
<td>40.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the extension.</td>
<td>40.0</td>
</tr>
<tr>
<td>information</td>
<td>List&lt;ConnectApi.AbstractExtensionInformation&gt;</td>
<td>Collection of extension information.</td>
<td>40.0</td>
</tr>
<tr>
<td>isEnabled InCommunity</td>
<td>Boolean</td>
<td>Indicates whether the extension is enabled in the site.</td>
<td>40.0 only</td>
</tr>
<tr>
<td>isEnabled InLightningPublisher</td>
<td>Boolean</td>
<td>Indicates whether the extension is enabled in the Lightning publisher.</td>
<td>40.0 only</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the extension.</td>
<td>40.0</td>
</tr>
<tr>
<td>position</td>
<td>Integer</td>
<td>Position in which the extension is displayed in the publisher.</td>
<td>41.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.ExtensionDefinitions

### ConnectApi.ExtensionDefinitions

A collection of extension definitions.

---

1783
### ConnectApi.ExtensionsCapability

If a feed element has this capability, it has one or more extension attachments.

Subclass of `ConnectApi.FeedElementCapability`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>items</td>
<td>List&lt;ConnectApi.Extension&gt;</td>
<td>List of extensions associated with the feed element.</td>
<td>40.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

- `ConnectApi.FeedElementCapabilities`

### ConnectApi.ExternalCredential

External credential, including the named credentials and principals associated with it and the type and status of each principal.

If you don’t have the View Setup and Configuration permission, some properties are empty or show limited information.

**Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>authenticationProtocol</td>
<td>ConnectApi.CredentialAuthenticationProtocol</td>
<td>Authentication protocol of the external credential.</td>
<td>56.0</td>
</tr>
</tbody>
</table>

Values are:
- AwsSv4
- Custom
- Jwt
- OAuth
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>authenticationStatus</td>
<td>ConnectApi.CredentialAuthenticationStatus</td>
<td>Status of the credential authentication. Values are:   - Configured—Credential has all required credentials for at least one principal. - NotConfigured—Credential isn’t configured. - Unknown—Credential status can’t be determined because the authentication protocol is custom.</td>
<td>56.0</td>
</tr>
<tr>
<td>createdByNamespace</td>
<td>String</td>
<td>Namespace of the package that created the external credential.</td>
<td>59.0</td>
</tr>
<tr>
<td>customHeaders</td>
<td>List&lt;ConnectApi.CredentialCustomHeader&gt;</td>
<td>List of custom headers.</td>
<td>57.0</td>
</tr>
<tr>
<td>developerName</td>
<td>String</td>
<td>Fully qualified developer name of the external credential.</td>
<td>56.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>External credential ID.</td>
<td>58.0</td>
</tr>
<tr>
<td>masterLabel</td>
<td>String</td>
<td>External credential label.</td>
<td>56.0</td>
</tr>
<tr>
<td>parameters</td>
<td>List&lt;ConnectApi.ExternalCredentialParameter&gt;</td>
<td>List of parameters of the external credential.</td>
<td>57.0</td>
</tr>
<tr>
<td>principals</td>
<td>List&lt;ConnectApi.ExternalCredentialPrincipal&gt;</td>
<td>List of principals the credential has.</td>
<td>56.0</td>
</tr>
<tr>
<td>relatedNamedCredentials</td>
<td>List&lt;ConnectApi.NamedCredential&gt;</td>
<td>List of named credentials associated to the external credential.</td>
<td>56.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>--------------</td>
<td>---------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>Connect REST API URL for the external credential.</td>
<td>58.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.ExternalCredentialList
- ConnectApi.NamedCredential

### ConnectApi.ExternalCredentialList
List of external credentials.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>externalCredentials</td>
<td>List&lt;ConnectApi.ExternalCredential&gt;</td>
<td>List of external credentials.</td>
<td>56.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ExternalCredentialParameter
External credential parameter.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>Parameter ID.</td>
<td>58.0</td>
</tr>
<tr>
<td>parameterDescription</td>
<td>String</td>
<td>Parameter description.</td>
<td>58.0</td>
</tr>
<tr>
<td>parameterName</td>
<td>String</td>
<td>Parameter name. If the parameterType is AuthParameter, valid values are:</td>
<td>57.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- AwsAccountId—Valid for AwsSv4.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- AwsProfileArn—Valid for AwsSv4 with RolesAnywhere.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- AwsRegion—Valid for AwsSv4.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- AwsService—Valid for AwsSv4.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- AwsStsDuration—Valid for AwsSv4 with STS or RolesAnywhere.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- AwsStsExternalId—Valid for AwsSv4 with STS.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- AwsTrustAnchorArn—Valid for AwsSv4 with RolesAnywhere.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Scope—Valid for OAuth.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other parameter types can be any value.</td>
<td></td>
</tr>
<tr>
<td>parameterType</td>
<td>ConnectApi.ExternalCredentialParameterType</td>
<td>Parameter type of the external credential. Values are:</td>
<td>57.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- AuthParameter</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- AuthProvider</td>
<td></td>
</tr>
</tbody>
</table>
### Available Version

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>authProviderUrl</td>
<td></td>
<td></td>
<td>57.0</td>
</tr>
<tr>
<td>authProviderUrlQueryParameter</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>jwtBodyClaim</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>jwtHeaderClaim</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>signingCertificate</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**parameterValue** | **String** | Parameter value. If the parameterType is JwtBodyClaim or JwtHeaderClaim, the parameter value can contain formulas. If the parameterType is AuthProvider or SigningCertificate, the parameter value is the fully qualified entity name of the corresponding entity. 57.0

**SEE ALSO:**
- ConnectApi.ExternalCredential

### ConnectApi.ExternalCredentialPrincipal

External credential principal.

If you don’t have the View Setup and Configuration permission, some properties are empty or show limited information.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>authenticationStatus</td>
<td>ConnectApi.</td>
<td>Status of the credential authentication. Values are:</td>
<td>56.0</td>
</tr>
<tr>
<td></td>
<td>CredentialAuthentication</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Status</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Configured—Credential has all required credentials for at least one principal.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>NotConfigured—Credential isn’t configured.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Unknown—Credential status can’t be determined because the authentication protocol is custom.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>id</td>
<td><strong>String</strong></td>
<td>ID of the external credential principal.</td>
<td>58.0</td>
</tr>
<tr>
<td>parameters</td>
<td>List&lt;ConnectApi.</td>
<td>List of external credential parameters.</td>
<td>58.0</td>
</tr>
<tr>
<td></td>
<td>ExternalCredentialParameter&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>principalAccess</td>
<td>List&lt;ConnectApi.</td>
<td>List of access entities associated with the external credential principal.</td>
<td>58.0</td>
</tr>
<tr>
<td></td>
<td>ExternalCredentialPrincipalAccess&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>principalName</td>
<td><strong>String</strong></td>
<td>Name of the external credential named principal.</td>
<td>56.0</td>
</tr>
</tbody>
</table>
### ConnectApi.ExternalCredentialPrincipal

External credential principal.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
</table>
| principalType | ConnectApi.CredentialPrincipalType | Type of credential principal. Values are:  
  - AwsStsPrincipal  
  - NamedPrincipal  
  - PerUserPrincipal | 56.0               |
| sequenceNumber| Integer           | Sequence number of the external credential principal.                        | 58.0               |

**SEE ALSO:**

- ConnectApi.ExternalCredential

### ConnectApi.ExternalCredentialPrincipalAccess

External credential principal access.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>developerName</td>
<td>String</td>
<td>Developer name of the associated access entity.</td>
<td>58.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the associated access entity.</td>
<td>58.0</td>
</tr>
</tbody>
</table>
| type                | ConnectApi.ExternalCredentialPrincipalAccessType | Access type of the external credential principal. Values are:  
  - PermissionSet  
  - PermissionSetGroup  
  - Profile          | 58.0               |

**SEE ALSO:**

- ConnectApi.ExternalCredentialPrincipal

### ConnectApi.ExternalFilePermissionInformation

External file permission information.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>external FilePermissionTypes</td>
<td>List&lt;ConnectApi.ContentHubPermissionType&gt;</td>
<td>Available permission types for the parent folder of the external file, or null for non-external files or when includeExternalFilePermissionsInfo is false.</td>
<td>39.0</td>
</tr>
<tr>
<td>external FilePermissionsFailure</td>
<td>Boolean</td>
<td>true if the retrieval of external file information failed or if includeExternalFilePermissionsInfo is false; false otherwise.</td>
<td>39.0</td>
</tr>
</tbody>
</table>
### Available Version

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>external FilePermissions</td>
<td>String</td>
<td>Explanation of the failure if a failure occurred and includeExternalFilePermissionsInfo is true; null otherwise.</td>
<td>39.0</td>
</tr>
<tr>
<td>InfoFailureReason</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>external FileSharingStatus</td>
<td>ConnectApi.ContentHub</td>
<td>Sharing status for the external file. Values are:</td>
<td>39.0</td>
</tr>
<tr>
<td></td>
<td>ExternalItem SharingType</td>
<td>• DomainSharing—File is shared with the domain.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• PrivateSharing—File is private or shared only with individuals.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• PublicSharing—File is publicly shared.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Value is null for non-external files or when includeExternalFilePermissionsInfo is false.</td>
<td></td>
</tr>
<tr>
<td>repository PublicGroups</td>
<td>List&lt;ConnectApi.RepositoryGroupSummary&gt;</td>
<td>Available public groups in the external repository or null for non-external files or when includeExternalFilePermissionsInfo is false.</td>
<td>39.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

- ConnectApi.AbstractRepositoryFile

---

### ConnectApi.ExternalManagedAccountAddressOutput

Default shipping address for an externally managed account.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>city</td>
<td>String</td>
<td>City of the external managed account record.</td>
<td>53.0</td>
</tr>
<tr>
<td>country</td>
<td>String</td>
<td>Country of the external managed account record.</td>
<td>53.0</td>
</tr>
<tr>
<td>geolocationAccuracy</td>
<td>String</td>
<td>Geolocation accuracy of the external managed account record.</td>
<td>53.0</td>
</tr>
<tr>
<td>latitude</td>
<td>String</td>
<td>Latitude of the external managed account record.</td>
<td>53.0</td>
</tr>
<tr>
<td>longitude</td>
<td>String</td>
<td>Longitude of the external managed account record.</td>
<td>53.0</td>
</tr>
<tr>
<td>state</td>
<td>String</td>
<td>State of the external managed account record.</td>
<td>53.0</td>
</tr>
<tr>
<td>street</td>
<td>String</td>
<td>Street of the external managed account record.</td>
<td>53.0</td>
</tr>
<tr>
<td>zip</td>
<td>String</td>
<td>Postal code of the external managed account record.</td>
<td>53.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

- ConnectApi.ExternalManagedAccountOutput
ConnectApi.ExternalManagedAccountCollectionOutput

Collection of externally managed accounts.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>externalManagedAccounts</td>
<td>List&lt;ConnectApi.ExternalManagedAccountOutput&gt;</td>
<td>Collection of externally managed accounts.</td>
<td>49.0</td>
</tr>
<tr>
<td>totalExternalManagedAccounts</td>
<td>Integer</td>
<td>Total number of externally managed accounts.</td>
<td>49.0</td>
</tr>
</tbody>
</table>

ConnectApi.ExternalManagedAccountOutput

Externally managed account.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>accountId</td>
<td>String</td>
<td>ID of the account managed by another account.</td>
<td>49.0</td>
</tr>
<tr>
<td>accountName</td>
<td>String</td>
<td>Name of the external managed account record.</td>
<td>53.0</td>
</tr>
<tr>
<td>address</td>
<td>ConnectApi.ExternalManagedAccountAddressOutput</td>
<td>Default shipping address of the external managed account.</td>
<td>53.0</td>
</tr>
<tr>
<td>externalManagedAccountId</td>
<td>String</td>
<td>ID of the external managed account record.</td>
<td>49.0</td>
</tr>
<tr>
<td>isMyAccount</td>
<td>Boolean</td>
<td>Specifies whether the account is the context user's account (true) or not (false).</td>
<td>53.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

ConnectApi.ExternalManagedAccountCollectionOutput

ConnectApi.FacetValue

Facet value.

This class is abstract and is a superclass of ConnectApi.DistinctFacetValue.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>ConnectApi.CommerceSearchFacetType</td>
<td>Search facet type. Value is:</td>
<td>52.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• DistinctValue</td>
<td></td>
</tr>
</tbody>
</table>

ConnectApi.Features

Features available to the context user in the org.
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>activityReminder</td>
<td>Boolean</td>
<td>Reserved for future use.</td>
<td>37.0</td>
</tr>
<tr>
<td>Notifications Enabled</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>chatter</td>
<td>Boolean</td>
<td>Specifies whether Chatter is enabled.</td>
<td>28.0</td>
</tr>
<tr>
<td>chatterActivity</td>
<td>Boolean</td>
<td>Specifies whether user details include information about Chatter activity.</td>
<td>28.0</td>
</tr>
<tr>
<td>chatterAnswers</td>
<td>Boolean</td>
<td>Specifies whether Chatter Answers is enabled.</td>
<td>29.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> With the Spring ’18 release, Salesforce no longer supports Chatter Answers. Users of Chatter Answers can post, answer, comment, or view existing Chatter Answers data, but support and updates are scheduled to end. We recommend transitioning to Chatter Questions. For more information, see End of Support for Chatter Answers in Spring ’18.</td>
<td></td>
</tr>
<tr>
<td>chatterGroup</td>
<td>Boolean</td>
<td>Specifies whether Chatter groups can have records associated with them.</td>
<td>30.0</td>
</tr>
<tr>
<td>Records</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>chatterGroupRecordSharing</td>
<td>Boolean</td>
<td>Specifies whether Chatter records are implicitly shared among group members when records are added to groups.</td>
<td>30.0</td>
</tr>
<tr>
<td>chatterMessages</td>
<td>Boolean</td>
<td>Specifies whether Chatter messages are enabled.</td>
<td>28.0</td>
</tr>
<tr>
<td>chatterTopics</td>
<td>Boolean</td>
<td>Specifies whether topics are enabled.</td>
<td>28.0</td>
</tr>
<tr>
<td>communitiesEnabled</td>
<td>Boolean</td>
<td>Specifies whether digital experiences is enabled.</td>
<td>31.0</td>
</tr>
<tr>
<td>communityModeration</td>
<td>Boolean</td>
<td>Specifies whether moderation is enabled.</td>
<td>29.0</td>
</tr>
<tr>
<td>communityReputation</td>
<td>Boolean</td>
<td>Specifies whether reputation is enabled.</td>
<td>32.0</td>
</tr>
<tr>
<td>dashboardComponentSnapshots</td>
<td>Boolean</td>
<td>Specifies whether the user can post dashboard component snapshots.</td>
<td>28.0</td>
</tr>
<tr>
<td>defaultCurrencyIsoCode</td>
<td>String</td>
<td>ISO code of the default currency. Applicable only when multiCurrency is false.</td>
<td>28.0</td>
</tr>
<tr>
<td>einsteinVoiceEnabled</td>
<td>Boolean</td>
<td>Reserved for future use.</td>
<td>46.0</td>
</tr>
<tr>
<td>Property</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>------------------------</td>
<td>--------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>eulerVoiceInPilotEnabled</td>
<td>Boolean</td>
<td>Reserved for future use.</td>
<td>46.0</td>
</tr>
<tr>
<td>eulerVoiceLoggingEnabled</td>
<td>Boolean</td>
<td>Reserved for future use.</td>
<td>46.0</td>
</tr>
<tr>
<td>eulerVoiceProviderId</td>
<td>Integer</td>
<td>Reserved for future use.</td>
<td>46.0</td>
</tr>
<tr>
<td>favoritesEnabled</td>
<td>Boolean</td>
<td>Specifies whether favorites in Lightning are enabled.</td>
<td>41.0</td>
</tr>
<tr>
<td>feedPolling</td>
<td>Boolean</td>
<td>Reserved for future use.</td>
<td>28.0</td>
</tr>
<tr>
<td>feedStreamEnabled</td>
<td>Boolean</td>
<td>Specifies whether Chatter feed streams are enabled.</td>
<td>39.0</td>
</tr>
<tr>
<td>files</td>
<td>Boolean</td>
<td>Specifies whether files can act as resources for Connect REST API.</td>
<td>28.0</td>
</tr>
<tr>
<td>filesOnComments</td>
<td>Boolean</td>
<td>Specifies whether files can be attached to comments.</td>
<td>28.0</td>
</tr>
<tr>
<td>forecasting3AggregatedEnabled</td>
<td>Boolean</td>
<td>Specifies whether aggregated forecasting is enabled for mobile clients.</td>
<td>38.0</td>
</tr>
<tr>
<td>forecastingEnabled</td>
<td>Boolean</td>
<td>Specifies whether forecasting is enabled.</td>
<td>38.0</td>
</tr>
<tr>
<td>forecastingPeriodRange</td>
<td>Integer</td>
<td>Range of the forecasting period.</td>
<td>38.0</td>
</tr>
<tr>
<td>forecastingPeriodStart</td>
<td>Integer</td>
<td>Start index for the forecasting period.</td>
<td>38.0</td>
</tr>
<tr>
<td>forecastingPeriodType</td>
<td>ConnectApi.PeriodType</td>
<td>Time period used for forecasting. Values are:</td>
<td>38.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Month</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Quarter</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Week</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Year</td>
<td></td>
</tr>
<tr>
<td>groupsCanFollow</td>
<td>Boolean</td>
<td>Reserved for future use.</td>
<td>28.0–29.0</td>
</tr>
<tr>
<td>ideas</td>
<td>Boolean</td>
<td>Specifies whether Ideas is enabled.</td>
<td>29.0</td>
</tr>
<tr>
<td>liveAgentHostName</td>
<td>String</td>
<td>Live Agent host name configured for the org.</td>
<td>41.0</td>
</tr>
<tr>
<td>managedTopicsEnabled</td>
<td>Boolean</td>
<td>Specifies whether managed topics are enabled.</td>
<td>32.0</td>
</tr>
<tr>
<td>Property</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>maxEntity Subscriptions PerStream</td>
<td>Integer</td>
<td>Specifies the maximum number of feed-enabled entities that can be subscribed to in a Chatter stream.</td>
<td>39.0</td>
</tr>
<tr>
<td>maxFiles PerFeedItem</td>
<td>Integer</td>
<td>Specifies the maximum number of files that can be added to a feed item.</td>
<td>36.0</td>
</tr>
<tr>
<td>maxStreams PerPerson</td>
<td>Integer</td>
<td>Specifies the maximum number of Chatter streams that a user can have.</td>
<td>39.0</td>
</tr>
<tr>
<td>mobile Notifications Enabled</td>
<td>Boolean</td>
<td>Reserved for future use.</td>
<td>29.0</td>
</tr>
<tr>
<td>multiCurrency</td>
<td>Boolean</td>
<td>Specifies whether the org uses multiple currencies (true) or not (false). When false, the defaultCurrencyIsoCode indicates the ISO code of the default currency.</td>
<td>28.0</td>
</tr>
<tr>
<td>offlineEdit Enabled</td>
<td>Boolean</td>
<td>Specifies whether the offline object permissions are enabled for Salesforce for Android and Salesforce for iOS mobile clients.</td>
<td>37.0</td>
</tr>
<tr>
<td>publisherActions</td>
<td>Boolean</td>
<td>Specifies whether actions in the publisher are enabled.</td>
<td>28.0</td>
</tr>
<tr>
<td>storeData OnDevices Enabled</td>
<td>Boolean</td>
<td>Specifies whether the Salesforce for Android and Salesforce for iOS can use secure, persistent storage on mobile devices to cache data.</td>
<td>30.0</td>
</tr>
<tr>
<td>thanksAllowed</td>
<td>Boolean</td>
<td>Reserved for future use.</td>
<td>28.0</td>
</tr>
<tr>
<td>trendingTopics</td>
<td>Boolean</td>
<td>Specifies whether trending topics are enabled.</td>
<td>28.0</td>
</tr>
<tr>
<td>userNav ItemsEnabled</td>
<td>Boolean</td>
<td>Specifies whether users can customize the navigation bar in Lightning.</td>
<td>41.0</td>
</tr>
<tr>
<td>viralInvites Allowed</td>
<td>Boolean</td>
<td>Specifies whether existing Chatter users can invite people in their company to use Chatter.</td>
<td>28.0</td>
</tr>
<tr>
<td>wave</td>
<td>Boolean</td>
<td>Specifies whether CRM Analytics is enabled.</td>
<td>36.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- getSettings()
- ConnectApi.OrganizationSettings

**ConnectApi.Feed**

Chatter feed.
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>feedElement</td>
<td>String</td>
<td>Connect REST API URL to post feed elements to this subject.</td>
<td>31.0</td>
</tr>
<tr>
<td>PostUrl</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>feedElements</td>
<td>ConnectApi.FeedElements</td>
<td>Page of feed elements for the feed specified in redirectedFeedType. Otherwise, null.</td>
<td>40.0</td>
</tr>
<tr>
<td>feedElementsUrl</td>
<td>String</td>
<td>Connect REST API URL to feed elements.</td>
<td>31.0</td>
</tr>
<tr>
<td>feedItemsUrl</td>
<td>String</td>
<td>Connect REST API URL to feed items.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>isModifedUrl</td>
<td>String</td>
<td>Connect REST API URL with a since request parameter that contains an opaque token that describes when the feed was last modified. Returns null if the feed isn’t a news feed. Use this URL to poll a news feed for updates. Important: This feature is available through a Feed Polling pilot program. This pilot program is closed and not accepting new participants.</td>
<td>28.0</td>
</tr>
<tr>
<td>pinnedFeedElementsUrl</td>
<td>String</td>
<td>URL to pinned feed items.</td>
<td>41.0</td>
</tr>
<tr>
<td>redirectedFeedFilter</td>
<td>ConnectApi.FeedFilter</td>
<td>Filter for the feed specified in redirectedFeedType. Otherwise, null.</td>
<td>42.0</td>
</tr>
<tr>
<td>redirectedFeedSort</td>
<td>ConnectApi.FeedSort</td>
<td>Sort order for the feed specified in redirectedFeedType. Otherwise, null.</td>
<td>42.0</td>
</tr>
<tr>
<td>redirectedFeedType</td>
<td>ConnectApi.FeedType</td>
<td>Specifies which feed is returned if pageSize is specified. Otherwise, null.</td>
<td>40.0</td>
</tr>
<tr>
<td>respectsMute</td>
<td>Boolean</td>
<td>Indicates whether the feed respects the mute feature. If true, the feed shows the ability to mute or unmute each element, depending on the value of isMutedByMe; null if the mute feature is disabled for the organization.</td>
<td>35.0</td>
</tr>
</tbody>
</table>

**ConnectApi.FeedBody**

Feed body.


No additional properties.

SEE ALSO:

ConnectApi.Comment

ConnectApi.FeedElement

ConnectApi.FeedEntitySummary
### ConnectApi.FeedDirectory

Directory of feeds and favorites.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>favorites</td>
<td>List&lt;ConnectApi.FeedFavorite&gt;</td>
<td>A list of feed favorites</td>
<td>30.0</td>
</tr>
<tr>
<td>feeds</td>
<td>List&lt;ConnectApi.FeedDirectoryItem&gt;</td>
<td>A list of feeds</td>
<td>30.0</td>
</tr>
</tbody>
</table>

### ConnectApi.FeedDirectoryItem

Definition of a feed.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>feedElementsUrl</td>
<td>String</td>
<td>Connect REST API resource URL for the feed elements.</td>
<td></td>
</tr>
<tr>
<td>feedItemsUrl</td>
<td>String</td>
<td>Connect REST API resource URL for the feed items of a specific feed.</td>
<td>30.0–31.0</td>
</tr>
<tr>
<td>feedType</td>
<td>ConnectApi.FeedTypeEnum</td>
<td>The feed type. One of these values:</td>
<td>30.0</td>
</tr>
<tr>
<td></td>
<td>• Bookmarks—Contains all feed items saved as bookmarks by the context user.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Company—Contains all feed items except feed items of type TrackedChange. To see the feed item, the user must have sharing access to its parent.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• DirectMessageModeration—Contains all direct messages that are flagged for moderation. The Direct Message Moderation feed is available only to users with Moderate Experiences Chatter Messages permissions.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• DirectMessages—Contains all feed items of the context user's direct messages.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Draft—Contains all the feed items that the context user drafted.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Files—Contains all feed items that contain files posted by people or groups that the context user follows.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Filter—Contains the news feed filtered to contain feed items whose parent is a specified object type.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Groups—Contains all feed items from all groups the context user either owns or is a member of.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Home—Contains all feed items associated with any managed topic in an Experience Cloud site.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Landing—Contains all feed items that best drive user engagement when the feed is requested. Allows clients to avoid an empty feed when there aren't many personalized feed items.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
- **Moderation**—Contains all feed items that are flagged for moderation, except direct messages. The moderation feed is available only to users with Moderate Experiences Feeds permissions.
- **Mute**—Contains all feed items that the context user muted.
- **News**—Contains all updates for people the context user follows, groups the user is a member of, and files and records the user is following. Contains all updates for records whose parent is the context user.
- **PendingReview**—Contains all feed items and comments that are pending review.
- **People**—Contains all feed items posted by all people the context user follows.
- **Record**—Contains all feed items whose parent is a specified record, which could be a group, user, object, file, or any other standard or custom object. When the record is a group, the feed also contains feed items that mention the group. When the record is a user, the feed contains only feed items on that user. You can get another user's record feed.
- **Streams**—Contains all feed items for any combination of up to 25 feed-enabled entities that the context user subscribes to in a stream. Examples of feed-enabled entities include people, groups, and records.
- **To**—Contains all feed items with mentions of the context user. Contains feed items the context user commented on and feed items created by the context user that are commented on.
- **Topics**—Contains all feed items that include the specified topic.
- **UserProfile**—Contains feed items created when a user changes records that can be tracked in a feed. Contains feed items whose parent is the user and feed items that @mention the user. This feed is different than the news feed, which returns more feed items, including group updates. You can get another user's user profile feed.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>feedUrl</td>
<td>String</td>
<td>Connect REST API resource URL for a specific feed</td>
</tr>
<tr>
<td>keyPrefix</td>
<td>String</td>
<td>A key prefix is the first three characters of a record ID, which specifies the object type. For filter feeds, this value is the key prefix associated with the object type used to filter this feed. All feed items in this feed have a parent whose object type matches this key prefix value. For non-filter feeds, this value is null.</td>
</tr>
</tbody>
</table>
## ConnectApi.FeedElement

Feed elements are the top-level items that a feed contains. Feeds are feed element containers.

This class is abstract.

Superclass of:
- ConnectApi.FeedItem
- ConnectApi.GenericFeedElement

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>body</td>
<td>ConnectApi.FeedBody</td>
<td>Information about the feed element.</td>
<td>22.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Important: Use the header.text property as the default value for rendering text because the body.text property can be <strong>null</strong>.</td>
<td></td>
</tr>
<tr>
<td>capabilities</td>
<td>ConnectApi.FeedElementCapabilities</td>
<td>A container for all capabilities that can be included with a feed element.</td>
<td>31.0</td>
</tr>
<tr>
<td>createdDate</td>
<td>Datetime</td>
<td>ISO 8601 format date string, for example, 2011-02-25T18:24:31.000Z.</td>
<td>31.0</td>
</tr>
<tr>
<td>feedElementType</td>
<td>ConnectApi.FeedElementType</td>
<td>Feed elements are the top-level objects that a feed contains. The feed element type describes the characteristics of that feed element. One of these values:</td>
<td>31.0</td>
</tr>
</tbody>
</table>

- **Bundle**—A container of feed elements. A bundle also has a body made up of message segments that can always be gracefully degraded to text-only values.
- **FeedItem**—A feed item has a single parent and is scoped to one Experience Cloud site or across all Experience Cloud sites. A feed item can have capabilities such as bookmarks, canvas, content, comment, link, poll. Feed items have a body made up of message segments that can always be gracefully degraded to text-only values.
### Recommendation

- **Recommendation**—A recommendation is a feed element with a recommendations capability. A recommendation suggests records to follow, groups to join, or applications that are helpful to the context user.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>header</td>
<td>ConnectApi.MessageBody</td>
<td>The header is the title of the post. This property contains renderable plain text for all the segments of the message. If a client doesn’t know how to render a feed element type, it should render this text.</td>
<td>31.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>18-character ID of the feed element.</td>
<td>22.0</td>
</tr>
<tr>
<td>modifiedDate</td>
<td>Datetime</td>
<td>ISO 8601 format date string, for example, 2011-02-25T18:24:31.000Z.</td>
<td>31.0</td>
</tr>
<tr>
<td>parent</td>
<td>ConnectApi.ActorWithId</td>
<td>Feed element’s parent</td>
<td>28.0</td>
</tr>
<tr>
<td>relativeCreatedDate</td>
<td>String</td>
<td>The created date formatted as a relative, localized string, for example, “17m ago” or “Yesterday.”</td>
<td>31.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>Connect REST API URL to this feed element.</td>
<td>22.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.Announcement
- ConnectApi.FeedElementPage
- ConnectApi.PinnedFeedElements
- ConnectApi.QuestionAndAnswersSuggestions

### ConnectApi.FeedElementCapabilities

A container for all capabilities that can be included with a feed element.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>approval</td>
<td>ConnectApi.ApprovalCapability</td>
<td>If a feed element has this capability, it includes information about an approval.</td>
<td>32.0</td>
</tr>
<tr>
<td>associated Actions</td>
<td>ConnectApi.AssociatedActionsCapability</td>
<td>If a feed element has this capability, it has platform actions associated with it.</td>
<td>33.0</td>
</tr>
<tr>
<td>banner</td>
<td>ConnectApi.BannerCapability</td>
<td>If a feed element has this capability, it has a banner motif and style.</td>
<td>31.0</td>
</tr>
<tr>
<td>bookmarks</td>
<td>ConnectApi.BookmarksCapability</td>
<td>If a feed element has this capability, the context user can bookmark it.</td>
<td>31.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>------------------</td>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>bundle</td>
<td><code>ConnectApi.BundleCapability</code></td>
<td>If a feed element has this capability, it has a container of feed elements called a <strong>bundle</strong>.</td>
<td>31.0</td>
</tr>
<tr>
<td>callCollaboration</td>
<td><code>ConnectApi.CallCollaborationCapability</code></td>
<td>If a feed element has this capability, it has a recording comment.</td>
<td>51.0</td>
</tr>
<tr>
<td>canvas</td>
<td><code>ConnectApi.CanvasCapability</code></td>
<td>If a feed element has this capability, it renders a canvas app.</td>
<td>32.0</td>
</tr>
<tr>
<td>caseComment</td>
<td><code>ConnectApi.CaseCommentCapability</code></td>
<td>If a feed element has this capability, it has a case comment on the case feed.</td>
<td>32.0</td>
</tr>
<tr>
<td>chatterLikes</td>
<td><code>ConnectApi.ChatterLikesCapability</code></td>
<td>If a feed element has this capability, the context user can like it. Exposes information about existing likes.</td>
<td>31.0</td>
</tr>
<tr>
<td>close</td>
<td><code>ConnectApi.CloseCapability</code></td>
<td>If a feed element has this capability, users with permission can close it.</td>
<td>43.0</td>
</tr>
<tr>
<td>comments</td>
<td><code>ConnectApi.CommentsCapability</code></td>
<td>If a feed element or comment has this capability, the context user can add a comment to it.</td>
<td>31.0</td>
</tr>
</tbody>
</table>
| content          | `ConnectApi.ContentCapability` | If a comment has this capability, it has a file attachment.  
Most `ConnectApi.ContentCapability` properties are null if the content has been deleted from the feed element or if the access has changed to private. | 32.0–35.0         |
<p>| dashboardComponentSnapshot | <code>ConnectApi.DashboardComponentSnapshotCapability</code> | If a feed element has this capability, it has a dashboard component snapshot. A snapshot is a static image of a dashboard component at a specific point in time. | 32.0              |
| directMessage    | <code>ConnectApi.DirectMessageCapability</code> | If a feed element has this capability, it’s a direct message. | 39.0              |
| edit             | <code>ConnectApi.EditCapability</code> | If a feed element has this capability, users who have permission can edit it. | 34.0              |
| emailMessage     | <code>ConnectApi.EmailMessageCapability</code> | If a feed element has this capability, it has an email message from a case. | 32.0              |</p>
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>enhancedLink</td>
<td>ConnectApi. EnhancedLink Capability</td>
<td>If a feed element has this capability, it has a link that may contain</td>
<td>32.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>supplemental information like an icon, a title, and a description.</td>
<td></td>
</tr>
<tr>
<td>extensions</td>
<td>ConnectApi. ExtensionsCapability</td>
<td>If a feed element has this capability, it has one or more extension</td>
<td>40.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>attachments.</td>
<td></td>
</tr>
<tr>
<td>feedEntityShare</td>
<td>ConnectApi. FeedShare Capability</td>
<td>If a feed element or comment has this capability, a feed entity is</td>
<td>39.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>shared with it.</td>
<td></td>
</tr>
<tr>
<td>files</td>
<td>ConnectApi. FilesCapability</td>
<td>If a feed element has this capability, it has one or more file attachments.</td>
<td>36.0</td>
</tr>
<tr>
<td>interactions</td>
<td>ConnectApi. InteractionsCapability</td>
<td>If a feed element has this capability, it has information about user</td>
<td>37.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>interactions.</td>
<td></td>
</tr>
<tr>
<td>link</td>
<td>ConnectApi. LinkCapability</td>
<td>If a feed element has this capability, it has a link.</td>
<td>32.0</td>
</tr>
<tr>
<td>mediaReferences</td>
<td>ConnectApi.MediaReferenceCapability</td>
<td>If a feed element has this capability, it has one or more media references.</td>
<td>41.0</td>
</tr>
<tr>
<td>moderation</td>
<td>ConnectApi. ModerationCapability</td>
<td>If a feed element has this capability, users in an Experience Cloud site</td>
<td>31.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>can flag it for moderation.</td>
<td></td>
</tr>
<tr>
<td>mute</td>
<td>ConnectApi. MuteCapability</td>
<td>If a feed element has this capability, users can mute it.</td>
<td>35.0</td>
</tr>
<tr>
<td>origin</td>
<td>ConnectApi. OriginCapability</td>
<td>If a feed element has this capability, it was created by a feed action.</td>
<td>33.0</td>
</tr>
<tr>
<td>pin</td>
<td>ConnectApi. PinCapability</td>
<td>If a feed element has this capability, users who have permission can pin</td>
<td>41.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>it to a feed.</td>
<td></td>
</tr>
<tr>
<td>poll</td>
<td>ConnectApi. PollCapability</td>
<td>If a feed element has this capability, it includes a poll.</td>
<td>31.0</td>
</tr>
<tr>
<td>questionAndAnswers</td>
<td>ConnectApi. QuestionAndAnswersCapability</td>
<td>If a feed element has this capability, it has a question and comments on</td>
<td>31.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>the feed element are answers to the question.</td>
<td></td>
</tr>
<tr>
<td>readBy</td>
<td>ConnectApi. ReadByCapability</td>
<td>If a feed element has this capability, the context user can mark it</td>
<td>40.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>as read.</td>
<td></td>
</tr>
<tr>
<td>recommendations</td>
<td>ConnectApi. RecommendationsCapability</td>
<td>If a feed element has this capability, it has a recommendation.</td>
<td>32.0</td>
</tr>
<tr>
<td>recordSnapshot</td>
<td>ConnectApi. RecordSnapshotCapability</td>
<td>If a feed element has this capability, it contains all the snapshotted</td>
<td>32.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>fields of a record for a single create record event.</td>
<td></td>
</tr>
</tbody>
</table>
### ConnectApi.FeedElementCapability

A feed element capability, which defines the characteristics of a feed element.

In API version 30.0 and earlier, most feed items can have comments, likes, topics, and so on. In version 31.0 and later, every feed item (and feed element) can have a unique set of capabilities. If a capability property exists on a feed element, that capability is available, even if the capability property doesn’t have a value. For example, if the ChatterLikes capability property exists on a feed element (with or without a value), the context user can like that feed element. If the capability property doesn’t exist, it isn’t possible to like that feed element. A capability can also contain associated data. For example, the Moderation capability contains data about moderation flags.

This class is abstract.

This class is a superclass of:
- ConnectApi.AssociatedActionsCapability
- ConnectApi.ApprovalCapability
- ConnectApi.BannerCapability
- ConnectApi.BookmarksCapability
- ConnectApi.BundleCapability
- ConnectApi.CallCollaborationCapability
- ConnectApi.CanvasCapability
- ConnectApi.CaseCommentCapability
- ConnectApi.ChatterLikesCapability
- ConnectApi.CloseCapability
ConnectApi.FeedElementPage

A paged collection of ConnectApi.FeedElement objects.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageToken</td>
<td>String</td>
<td>Token identifying the current page.</td>
<td>31.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>31.0</td>
</tr>
<tr>
<td>elements</td>
<td>List&lt;ConnectApi.FeedElement&gt;</td>
<td>Collection of feed elements.</td>
<td>31.0</td>
</tr>
</tbody>
</table>

This class doesn't have any properties.
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isModifiedToken</td>
<td>String</td>
<td>Opaque polling token to use in the <code>since</code> parameter of the <code>ChatterFeeds.isModified</code> method. The token describes when the feed was last modified.</td>
<td>31.0</td>
</tr>
<tr>
<td>isModifiedUrl</td>
<td>String</td>
<td>Connect REST API URL with a <code>since</code> request parameter that contains an opaque token that describes when the feed was last modified. Returns <code>null</code> if the feed isn’t a news feed. Use this URL to poll a news feed for updates.</td>
<td>31.0</td>
</tr>
<tr>
<td>nextPageToken</td>
<td>String</td>
<td>Token identifying the next page, or <code>null</code> if there isn’t a next page.</td>
<td>31.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or <code>null</code> if there isn’t a next page.</td>
<td>31.0</td>
</tr>
<tr>
<td>updatesToken</td>
<td>String</td>
<td>A token to use in a request to the <code>ConnectApi.ChatterFeeds.getFeedElementsUpdatedSince</code> method.</td>
<td>31.0</td>
</tr>
<tr>
<td>updatesUrl</td>
<td>String</td>
<td>Connect REST API feed resource containing the feed elements that have been updated since the feed was refreshed. If the feed doesn’t support this feature, the value is <code>null</code>.</td>
<td>31.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- `ConnectApi.BundleCapability`
- `ConnectApi.Feed`

**ConnectApi.FeedEnabledEntity**

An entity that can have feeds associated with it.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>The 18-character ID of the record.</td>
<td>39.0</td>
</tr>
</tbody>
</table>
Available Version | Description | Type | Property Name
--- | --- | --- | ---
39.0 | Small, medium, and large icons indicating the record’s type. | ConnectApi.Motif | motif
39.0 | The localized name of the record. | String | name
39.0 | The type of the record. | String | type
39.0 | URL to the record. | String | url

SEE ALSO:
ConnectApi.ChatterStream

**ConnectApi.FeedEntityIsEditable**
Indicates if the context user can edit a feed element or comment.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>areAttachmentsEditableByMe</td>
<td>Boolean</td>
<td><strong>true</strong> if the context user can add and remove attachments on the feed element or comment, <strong>false</strong> otherwise.</td>
<td>36.0</td>
</tr>
<tr>
<td>feedEntityUrl</td>
<td>String</td>
<td>URL of the feed element or comment.</td>
<td>34.0</td>
</tr>
<tr>
<td>isEditableByMe</td>
<td>Boolean</td>
<td><strong>true</strong> if the context user can edit the feed element or comment, <strong>false</strong> otherwise.</td>
<td>34.0</td>
</tr>
</tbody>
</table>

**ConnectApi.FeedEntityNotAvailableSummary**
A summary when the feed entity isn’t available.
Subclass of ConnectApi.FeedEntitySummary.
No additional properties.

**ConnectApi.FeedEntityReadSummary**
Summary of the feed post or comment that was read.
Subclass of ConnectApi.UserFeedEntityActivitySummary.
No additional properties.

**ConnectApi.FeedEntityShareCapability**
If a feed element or comment has this capability, a feed entity is shared with it.
Subclass of ConnectApi.FeedElementCapability.
**ConnectApi.FeedEntitySummary**

The summary of a feed entity that is shared with a feed element.

This class is abstract.

Superclass of:
- ConnectApi.FeedItemSummary
- ConnectApi.FeedEntityNotAvailableSummary

---

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>feedEntity</td>
<td>ConnectApi.FeedEntitySummary</td>
<td>The summary of the feed entity that is shared with the feed element or comment.</td>
<td>39.0</td>
</tr>
<tr>
<td>actor</td>
<td>ConnectApi.Actor</td>
<td>Entity that created the feed entity.</td>
<td>39.0</td>
</tr>
<tr>
<td>body</td>
<td>ConnectApi.FeedBody</td>
<td>Information about the feed entity.</td>
<td>39.0</td>
</tr>
<tr>
<td>createdDate</td>
<td>Datetime</td>
<td>ISO 8601 date string, for example, 2011-02-25T18:24:31.000Z, when the entity was created.</td>
<td>39.0</td>
</tr>
<tr>
<td>feedElementType</td>
<td>ConnectApi.FeedElementType</td>
<td>Type of feed entity.</td>
<td>39.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>18-character ID of the feed entity.</td>
<td>39.0</td>
</tr>
</tbody>
</table>
### isEntityAvailable

**Type:** Boolean

**Description:** Specifies whether the entity is available. If `false`, either the user doesn’t have access to the entity or the entity was deleted.

**Available Version:** 39.0

### parent

**Type:** `ConnectApi.ActorWithId`

**Description:** Parent of the feed entity.

**Available Version:** 39.0

### relativeCreatedDate

**Type:** String

**Description:** Relative created date, for example, “2h ago.”

**Available Version:** 39.0

### url

**Type:** String

**Description:** URL to the feed entity.

**Available Version:** 39.0

---

**SEE ALSO:**

- `ConnectApi.FeedEntityShareCapability`

---

### ConnectApi.FeedFavorite

**Description:** Feed favorite.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>community</td>
<td><code>ConnectApi.Reference</code></td>
<td>Information about the Experience Cloud site that contains the favorite.</td>
<td>28.0</td>
</tr>
<tr>
<td>createdBy</td>
<td><code>ConnectApi.User Summary</code></td>
<td>Favorite’s creator.</td>
<td>28.0</td>
</tr>
<tr>
<td>feedUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the feed item for this favorite.</td>
<td>28.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>Favorite’s 18–character ID.</td>
<td>28.0</td>
</tr>
<tr>
<td>lastViewDate</td>
<td><code>Datet ime</code></td>
<td>ISO 8601 date string, for example, <code>2011-02-25T18:24:31.000Z</code>.</td>
<td>28.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Favorite’s name.</td>
<td>28.0</td>
</tr>
<tr>
<td>searchText</td>
<td>String</td>
<td>If the favorite is from a search, contains the search text, otherwise, an empty string.</td>
<td>28.0</td>
</tr>
<tr>
<td>target</td>
<td><code>ConnectApi.Reference</code></td>
<td>A reference to the topic if applicable, <code>null</code> otherwise.</td>
<td>28.0</td>
</tr>
<tr>
<td>type</td>
<td><code>ConnectApi.FeedFavoriteType Enum</code></td>
<td>An empty string or one of the following values:</td>
<td>28.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ListView</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Search</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Topic</td>
<td></td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>Connect REST API URL to this favorite.</td>
<td>28.0</td>
</tr>
</tbody>
</table>
### ConnectApi.Favorite

Feed favorites.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>favorites</td>
<td>List&lt;ConnectApi.FeedFavorite&gt;</td>
<td>Complete list of favorites.</td>
<td>28.0</td>
</tr>
<tr>
<td>total</td>
<td>Integer</td>
<td>Total number of favorites.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

### ConnectApi.FeedItem

Feed item.

Subclass of ConnectApi.FeedElement Class as of version 31.0.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>actor</td>
<td>ConnectApi.Actor</td>
<td>The entity that created the feed item.</td>
<td>28.0</td>
</tr>
<tr>
<td>attachment</td>
<td>ConnectApi.FeedItemAttachment</td>
<td>Information about the attachment. If there is no attachment, returns null.</td>
<td>28.0–31.0</td>
</tr>
</tbody>
</table>

⚠️ **Important:** As of version 32.0, use the inherited `capabilities` property.

| canShare    | Boolean                   | Indicates whether the feed item can be shared. If a feed item has multiple file attachments and at least one attachment has been deleted or is inaccessible, the feed item can’t be shared. The `canShare` value is incorrectly set to `true` in these cases. | 28.0–38.0         |

⚠️ **Important:** As of version 39.0, use the `isSharable` property.

<p>| clientInfo | ConnectApi.ClientInfo     | Information about the connected app used to authenticate the connection. | 28.0              |</p>
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>comments</td>
<td><code>ConnectApi.CommentPage</code></td>
<td>First page of comments for this feed item.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Important:</strong> As of version 32.0, use the inherited <code>capabilities.comments.page</code> property.</td>
<td></td>
</tr>
<tr>
<td>event</td>
<td><code>Boolean</code></td>
<td><code>true</code> if feed item is created due to an event change, <code>false</code> otherwise.</td>
<td>22.0</td>
</tr>
<tr>
<td>hasVerified Comment</td>
<td><code>Boolean</code></td>
<td><code>true</code> if the feed item has a verified comment, otherwise <code>false</code>.</td>
<td>41.0</td>
</tr>
<tr>
<td>isBookmarked ByCurrentUser</td>
<td><code>Boolean</code></td>
<td><code>true</code> if the context user has bookmarked this feed item, otherwise, <code>false</code>.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Important:</strong> As of version 32.0, use the inherited <code>capabilities.bookmarks.isBookmarkedByCurrentUser</code> property.</td>
<td></td>
</tr>
<tr>
<td>isDelete Restricted</td>
<td><code>Boolean</code></td>
<td>If this property is <code>true</code> the comment cannot be deleted by the context user. If it is <code>false</code>, it might be possible for the context user to delete the comment, but it is not guaranteed.</td>
<td>28.0</td>
</tr>
<tr>
<td>isLikedBy CurrentUser</td>
<td><code>Boolean</code></td>
<td><code>true</code> if the context user has liked this feed item, otherwise, <code>false</code>.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Important:</strong> As of version 32.0, use the inherited <code>capabilities.chatterLikes.isLikedByCurrentUser</code> property.</td>
<td></td>
</tr>
<tr>
<td>isSharable</td>
<td><code>Boolean</code></td>
<td>Indicates whether the feed item can be shared.</td>
<td>39.0</td>
</tr>
<tr>
<td>likes</td>
<td><code>ConnectApi.ChatterLikePage</code></td>
<td>First page of likes for this feed item.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Important:</strong> As of version 32.0, use the inherited <code>capabilities.chatterLikes.page</code> property.</td>
<td></td>
</tr>
<tr>
<td>likesMessage</td>
<td><code>ConnectApi.MessageBody</code></td>
<td>A message body the describes who likes the feed item.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Important:</strong> As of version 32.0, use the inherited <code>capabilities.chatterLikes.likesMessage</code> property.</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>moderationFlags</td>
<td>ConnectApi. ModerationFlags</td>
<td>Information about the moderation flags on a feed item. If ConnectApi.Features.communityModeration is false, this property is null.</td>
<td>29.0–30.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Important:</strong> As of version 31.0, use the inherited capabilities.moderation.moderationFlags property.</td>
<td></td>
</tr>
<tr>
<td>myLike</td>
<td>ConnectApi.Reference</td>
<td>If the context user has liked the feed item, this property is a reference to the specific like, otherwise, null.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Important:</strong> As of version 32.0, use the inherited capabilities.chatterLikes.myLike property.</td>
<td></td>
</tr>
<tr>
<td>originalFeedItem</td>
<td>ConnectApi.Reference</td>
<td>A reference to the original feed item if this feed item is a shared feed item, otherwise, null.</td>
<td>28.0</td>
</tr>
<tr>
<td>originalFeedItemActor</td>
<td>ConnectApi.Actor</td>
<td>If this feed item is a shared feed item, returns information about the original poster of the feed item, otherwise, returns null.</td>
<td>28.0</td>
</tr>
<tr>
<td>photoUrl</td>
<td>String</td>
<td>URL of the photo associated with the feed item</td>
<td>28.0</td>
</tr>
<tr>
<td>preamble</td>
<td>ConnectApi.MessageBody</td>
<td>A collection of message segments, including the unformatted text of the message that you can use as the title of a feed item. Message segments include name, link, and motif icon information for the actor that created the feed item.</td>
<td>28.0–30.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Important:</strong> For API versions 29.0 and 30.0, use the ConnectApi.FeedItem.preamble.text property as the default case to render text. For API versions 31.0 and later, use the ConnectApi.FeedElement.header.text property as the default case to render text.</td>
<td></td>
</tr>
<tr>
<td>topics</td>
<td>ConnectApi.FeedItemTopicPage</td>
<td>Topics for this feed item.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Important:</strong> As of version 31.0, use the inherited capabilities.topics.items property.</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>type</td>
<td>ConnectApi.FeedItemType</td>
<td>Type of feed item.</td>
<td></td>
</tr>
</tbody>
</table>

**Important:** As of API version 32.0, use the `capabilities` property to determine what can be done with a feed item. See *Working with Feeds and Feed Elements*.

One of these values:

- **ActivityEvent**—Feed item generated in Case Feed when an event or task associated with a parent record with a feed enabled is created or updated.
- **AdvancedTextPost**—A feed item with advanced text formatting, such as a group announcement post.
- **ApprovalPost**—Feed item with an approval capability. Approvers can act on the feed item parent.
- **AttachArticleEvent**—Feed item generated when an article is attached to a case in Case Feed.
- **BasicTemplateFeedItem**—Feed item with an enhanced link capability.
- **CallLogPost**—Feed item generated when a call log is saved to a case in Case Feed.
- **CanvasPost**—Feed item generated by a canvas app in the publisher or from Connect REST API or Connect in Apex. The post itself is a link to a canvas app.
- **CaseCommentPost**—Feed item generated when a case comment is saved in Case Feed.
- **ChangeStatusPost**—Feed item generated when the status of a case is changed in Case Feed.
- **ChatTranscriptionPost**—Feed item generated in Case Feed when a Live Agent chat transcript is saved to a case.
- **CollaborationGroupCreated**—Feed item generated when a new public group is created. Contains a link to the new group.
- **CollaborationGroupUnarchived**—Deprecated. Feed item generated when an archived group is activated.
- **ContentPost**—Feed item with a content capability.
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CreateRecordEvent</td>
<td>Feed item</td>
<td>Feed item that describes a record created in the publisher.</td>
</tr>
<tr>
<td>DashboardComponentAlert</td>
<td>Feed item</td>
<td>Feed item with a dashboard alert.</td>
</tr>
<tr>
<td>DashboardComponentSnapshot</td>
<td>Feed item</td>
<td>Feed item with a dashboard component snapshot capability.</td>
</tr>
<tr>
<td>EmailMessageEvent</td>
<td>Feed item</td>
<td>Feed item generated when an email is sent from a case in Case Feed.</td>
</tr>
<tr>
<td>FacebookPost</td>
<td>Feed item</td>
<td>Deprecated. Feed item generated when a Facebook post is created from a case in Case Feed.</td>
</tr>
<tr>
<td>LinkPost</td>
<td>Feed item with a link capability.</td>
<td>Viewers of the feed item are allowed to vote on the options in the poll.</td>
</tr>
<tr>
<td>MilestoneEvent</td>
<td>Feed item</td>
<td>Feed item generated when a case milestone is either completed or reaches a violation status. Contains a link to the case milestone.</td>
</tr>
<tr>
<td>PollPost</td>
<td>Feed item with a poll capability.</td>
<td>Viewers of the feed item are allowed to vote on the options in the poll.</td>
</tr>
<tr>
<td>ProfileSkillPost</td>
<td>Feed item</td>
<td>Feed item generated when a skill is added to a user's profile.</td>
</tr>
<tr>
<td>QuestionPost</td>
<td>Feed item</td>
<td>Feed item generated when a question is asked.</td>
</tr>
<tr>
<td>ReplyPost</td>
<td>Feed item generated by a Chatter Answers reply.</td>
<td></td>
</tr>
<tr>
<td>RypplePost</td>
<td>Feed item generated when a user posts thanks.</td>
<td></td>
</tr>
<tr>
<td>SocialPost</td>
<td>Feed item generated when a social post is created from a case in Case Feed.</td>
<td></td>
</tr>
<tr>
<td>TextPost</td>
<td>Feed item containing text only.</td>
<td></td>
</tr>
<tr>
<td>TrackedChange</td>
<td>Feed item created when one or more fields on a record have been changed.</td>
<td></td>
</tr>
<tr>
<td>UserStatus</td>
<td>Feed item</td>
<td>Deprecated. A user's post to their own profile.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>visibility</th>
<th>Type of users who can see a feed item.</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>AllUsers</td>
<td>Visibility is not limited to internal users.</td>
<td>28.0</td>
</tr>
</tbody>
</table>
ConnectApi.FeedItemSummary

A feed item summary.

Subclass of `ConnectApi.FeedEntitySummary`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>capabilities</td>
<td><code>ConnectApi.FeedElementCapabilities</code></td>
<td>Container for all capabilities that can be included with a feed item.</td>
<td>39.0</td>
</tr>
<tr>
<td>header</td>
<td><code>ConnectApi.MessageBody</code></td>
<td>Title of the post. This property contains renderable plain text for all the message segments. If a client doesn’t know how to render a feed element type, it should render this text.</td>
<td>39.0</td>
</tr>
<tr>
<td>modifiedDate</td>
<td><code>Datetime</code></td>
<td>When the feed item was modified in the form of an ISO 8601 date string, for example, 2011-02-25T18:24:31.000Z.</td>
<td>39.0</td>
</tr>
<tr>
<td>originalFeedItem</td>
<td><code>ConnectApi.Reference</code></td>
<td>Reference to the original feed item if this feed item is a shared feed item; otherwise, <code>null</code>.</td>
<td>39.0</td>
</tr>
<tr>
<td>originalFeedItemActor</td>
<td><code>ConnectApi.Actor</code></td>
<td>If this feed item is a shared feed item, information about the original poster of the feed item; otherwise, <code>null</code>.</td>
<td>39.0</td>
</tr>
<tr>
<td>photoUrl</td>
<td><code>String</code></td>
<td>URL of the photo associated with the feed item.</td>
<td>39.0</td>
</tr>
<tr>
<td>visibility</td>
<td><code>ConnectApi.FeedItemVisibility</code></td>
<td>Specifies who can see a feed item.</td>
<td>39.0</td>
</tr>
</tbody>
</table>

- `InternalUsers`—Visibility is limited to internal users.
- `AllUsers`—Visibility is not limited to internal users.

ConnectApi.FeedModifiedInfo

Feed modified information.

⚠️ **Important:** This feature is available through a Feed Polling pilot program. This pilot program is closed and not accepting new participants.
### isModified

`isModified` is a boolean that returns `true` if the news feed has been modified since the last time it was polled; `false` otherwise. Returns `null` if the feed is not a news feed.

**Type:** Boolean

**Available Version:** 28.0

### isModifiedToken

`isModifiedToken` is an opaque polling token to use in the `since` parameter of the `ChatterFeeds.isModified` method. The token describes when the feed was last modified.

**Type:** String

**Available Version:** 28.0

### nextPollUrl

`nextPollUrl` is a Connect REST API URL with a `since` request parameter that contains an opaque token that describes when the feed was last modified. Returns `null` if the feed isn’t a news feed. Use this URL to poll a news feed for updates.

**Type:** String

**Available Version:** 28.0

---

### ConnectApi.FeedPollChoice

Feed poll choice.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>Poll choice ID.</td>
<td>28.0</td>
</tr>
<tr>
<td>position</td>
<td>Integer</td>
<td>The location in the poll where this poll choice exists.</td>
<td>28.0</td>
</tr>
<tr>
<td>text</td>
<td>String</td>
<td>Label text associated with the poll choice.</td>
<td>28.0</td>
</tr>
<tr>
<td>voteCount</td>
<td>Integer</td>
<td>Total number of votes for this poll choice.</td>
<td>28.0</td>
</tr>
<tr>
<td>voteCountRatio</td>
<td>Double</td>
<td>The ratio of total number of votes for this poll choice to all votes cast in the poll. Multiply the ratio by 100 to get the percentage of votes cast for this poll choice.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

SEE ALSO: 
- [ConnectApi.PollCapability](#)

### ConnectApi.FeedPostSummary

Summary of the post.

Subclass of [ConnectApi.UserActivitySummary](#).

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>feedItemId</td>
<td>String</td>
<td>ID of the post.</td>
<td>42.0</td>
</tr>
</tbody>
</table>

### ConnectApi.FeedReadSummary

Summary of the feed that was read.
Subclass of `ConnectApi.UserActivitySummary`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>containerId</td>
<td>String</td>
<td>ID of the parent of the feed.</td>
<td>42.0</td>
</tr>
<tr>
<td>feedType</td>
<td><code>ConnectApi.FeedType</code></td>
<td>Type of feed.</td>
<td>42.0</td>
</tr>
</tbody>
</table>

- **Bookmarks**—Contains all feed items saved as bookmarks by the context user.
- **Company**—Contains all feed items except feed items of type `TrackedChange`. To see the feed item, the user must have sharing access to its parent.
- **DirectMessageModeration**—Contains all direct messages that are flagged for moderation. The Direct Message Moderation feed is available only to users with Moderate Experiences Chatter Messages permissions.
- **DirectMessages**—Contains all feed items of the context user’s direct messages.
- **Draft**—Contains all the feed items that the context user drafted.
- **Files**—Contains all feed items that contain files posted by people or groups that the context user follows.
- **Filter**—Contains the news feed filtered to contain feed items whose parent is a specified object type.
- **Groups**—Contains all feed items from all groups the context user either owns or is a member of.
- **Home**—Contains all feed items associated with any managed topic in an Experience Cloud site.
- **Landing**—Contains all feed items that best drive user engagement when the feed is requested. Allows clients to avoid an empty feed when there aren’t many personalized feed items.
- **Moderation**—Contains all feed items that are flagged for moderation, except direct messages. The moderation feed is available only to users with Moderate Experiences Feeds permissions.
- **Mute**—Contains all feed items that the context user muted.
- **News**—Contains all updates for people the context user follows, groups the user is a member of, and files and records the user is following.
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contains all updates for records whose parent is the context user.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PendingReview—Contains all feed items and comments that are pending review.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>People—Contains all feed items posted by all people the context user follows.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Record—Contains all feed items whose parent is a specified record, which could be a group, user, object, file, or any other standard or custom object. When the record is a group, the feed also contains feed items that mention the group. When the record is a user, the feed contains only feed items on that user. You can get another user’s record feed.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Streams—Contains all feed items for any combination of up to 25 feed-enabled entities that the context user subscribes to in a stream. Examples of feed-enabled entities include people, groups, and records.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>To—Contains all feed items with mentions of the context user. Contains feed items the context user commented on and feed items created by the context user that are commented on.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Topics—Contains all feed items that include the specified topic.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>UserProfile—Contains feed items created when a user changes records that can be tracked in a feed. Contains feed items whose parent is the user and feed items that @mention the user. This feed is different than the news feed, which returns more feed items, including group updates. You can get another user’s user profile feed.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**ConnectApi.FieldChangeSegment**

Field change segment.

Subclass of **ConnectApi.ComplexSegment**.

No additional properties.

SEE ALSO:  
**ConnectApi.MoreChangesSegment**
ConnectApi.FieldChangeNameSegment
Field change name segment.
Subclass of ConnectApi.MessageSegment.
No additional properties.

ConnectApi.FieldChangeValueSegment
Field change value segment.
Subclass of ConnectApi.MessageSegment.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>valueType</td>
<td>ConnectApi.FieldChangeValueTypeEnum</td>
<td>Value type of a field change.</td>
<td>28.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NewValue—A new value</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• OldValue—An old value</td>
<td></td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>URL value if the field change is to a URL field (such as a web address)</td>
<td>28.0</td>
</tr>
</tbody>
</table>

ConnectApi.FieldValue
Field's value in product search results.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>value</td>
<td>String</td>
<td>Value of the field.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.ProductSummary

ConnectApi.File
File.
This class is abstract.
Subclass of ConnectApi.ActorWithId.
Superclass of ConnectApi.FileSummary.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>checksum</td>
<td>String</td>
<td>MD5 checksum for the file.</td>
<td>28.0</td>
</tr>
<tr>
<td>content</td>
<td>Datetime</td>
<td>ISO 8601 format date string, for example, 2011-02-25T18:24:31.000Z. File-specific modified date, which is updated only for direct file operations, such as rename.</td>
<td>32.0</td>
</tr>
<tr>
<td>ModifiedDate</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>--------------------</td>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>contentSize</td>
<td>Integer</td>
<td>Modifications to the file from outside of Salesforce can update this date.</td>
<td>28.0</td>
</tr>
<tr>
<td>contentUrl</td>
<td>String</td>
<td>If the file is a link, returns the URL, otherwise, the string <strong>null</strong>.</td>
<td>28.0</td>
</tr>
<tr>
<td>createdDate</td>
<td>Datetime</td>
<td>ISO 8601 date string when the file was created.</td>
<td>41.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the file.</td>
<td>28.0</td>
</tr>
<tr>
<td>downloadUrl</td>
<td>String</td>
<td>URL to the file, that can be used for downloading the file.</td>
<td>28.0</td>
</tr>
<tr>
<td>fileExtension</td>
<td>String</td>
<td>Extension of the file.</td>
<td>28.0</td>
</tr>
<tr>
<td>fileType</td>
<td>String</td>
<td>Type of file, such as PDF, PowerPoint.</td>
<td>28.0</td>
</tr>
<tr>
<td>flashRendition</td>
<td>String</td>
<td>Specifies if a flash preview version of the file has been rendered.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

**Note:** Flash renditions were retired on July 16, 2021.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isFileAsset</td>
<td>Boolean</td>
<td>Specifies whether the file is an asset.</td>
<td>46.0</td>
</tr>
<tr>
<td>isInMyFileSync</td>
<td>Boolean</td>
<td><strong>true</strong> if the file is synced with Salesforce Files Sync; <strong>false</strong> otherwise.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

**Note:** Salesforce Files Sync was retired on May 25, 2018.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isMajorVersion</td>
<td>Boolean</td>
<td><strong>true</strong> if the file is a major version; <strong>false</strong> if the file is a minor version. Major versions can’t be replaced.</td>
<td>31.0</td>
</tr>
<tr>
<td>mimeType</td>
<td>String</td>
<td>File’s MIME type.</td>
<td>28.0</td>
</tr>
<tr>
<td>moderationFlags</td>
<td>ConnectApi.ModerationFlags</td>
<td>Information about the moderation flags on a file. If ConnectApi.Features.communityModeration is <strong>false</strong>, this property is <strong>null</strong>.</td>
<td>30.0</td>
</tr>
<tr>
<td>modifiedDate</td>
<td>Datetime</td>
<td>ISO 8601 format date string, for example, 2011-02-25T18:24:31.000Z. Modifications to the file from within Salesforce update this date.</td>
<td>28.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the file.</td>
<td>28.0</td>
</tr>
<tr>
<td>origin</td>
<td>String</td>
<td>Specifies the file source. Valid values are:</td>
<td>28.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Chatter—file came from Chatter</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Content—file came from content</td>
<td></td>
</tr>
<tr>
<td>owner</td>
<td>ConnectApi.UserSummary</td>
<td>File’s owner.</td>
<td>28.0</td>
</tr>
<tr>
<td>pdfRendition</td>
<td>String</td>
<td>Specifies if a PDF preview version of the file has been rendered.</td>
<td>28.0</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>publishStatus</td>
<td>ConnectApi. FilePublishStatus</td>
<td>Specifies the publish status of the file.</td>
<td>28.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• PendingAccess—File is pending publishing.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• PrivateAccess—File is private.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• PublicAccess—File is public.</td>
<td></td>
</tr>
<tr>
<td>renditionUrl</td>
<td>String</td>
<td>URL to the rendition for the file.</td>
<td>28.0</td>
</tr>
<tr>
<td>renditionUrl</td>
<td>String</td>
<td>URL to the 240 x 180 rendition resource for the file.</td>
<td>29.0</td>
</tr>
<tr>
<td>240By180</td>
<td></td>
<td>For shared files, renditions process asynchronously after upload. For private files, renditions process when the first file preview is requested, and aren’t available immediately after the file is uploaded.</td>
<td></td>
</tr>
<tr>
<td>renditionUrl</td>
<td>String</td>
<td>URL to the 720 x 480 rendition resource for the file.</td>
<td>29.0</td>
</tr>
<tr>
<td>720By480</td>
<td></td>
<td>For shared files, renditions process asynchronously after upload. For private files, renditions process when the first file preview is requested, and aren’t available immediately after the file is uploaded.</td>
<td></td>
</tr>
<tr>
<td>sharingOption</td>
<td>ConnectApi. FileSharingOption</td>
<td>Sharing option of the file. Values are:</td>
<td>35.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Allowed—Resharing of the file is allowed.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Restricted—Resharing of the file is restricted.</td>
<td></td>
</tr>
<tr>
<td>sharingPrivacy</td>
<td>ConnectApi. FileSharingPrivacy</td>
<td>Sharing privacy of a file. Values are:</td>
<td>41.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• None—File is visible to anyone with record access.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• PrivateOnRecords—File is private on records.</td>
<td></td>
</tr>
<tr>
<td>sharingRole</td>
<td>ConnectApi. FileSharingType</td>
<td>Sharing role of the file.</td>
<td>28.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Admin—Owner permission, but doesn’t own the file.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Collaborator—Viewer permission, and can edit, change permissions, and upload a new version of a file.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Owner—Collaborator permission, and can make a file private, and delete a file.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Viewer—Can view, download, and share a file.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• WorkspaceManaged—Permission controlled by the library.</td>
<td></td>
</tr>
<tr>
<td>systemModstamp</td>
<td>Datetime</td>
<td>ISO 8601 date string indicating when a user or any automated system process, such as a trigger, updated the file.</td>
<td>41.0</td>
</tr>
<tr>
<td>textPreview</td>
<td>String</td>
<td>Text preview of the file if available; null otherwise.</td>
<td>30.0</td>
</tr>
<tr>
<td>thumb120By90</td>
<td>String</td>
<td>Specifies the rendering status of the 120 x 90 preview image of the file. One of these values:</td>
<td>28.0</td>
</tr>
<tr>
<td>RenditionStatus</td>
<td></td>
<td>• Processing—Image is being rendered.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Failed—Rendering process failed.</td>
<td></td>
</tr>
</tbody>
</table>
### Available Version

<table>
<thead>
<tr>
<th>Available Version</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>28.0</td>
<td>Rendering status of the 240 x 180 preview image of the file. One of these values:</td>
</tr>
<tr>
<td></td>
<td>• Success—Rendering process was successful.</td>
</tr>
<tr>
<td></td>
<td>• Na—Rendering is not available for this image.</td>
</tr>
<tr>
<td>28.0</td>
<td>Rendering status of the 720 x 480 preview image of the file. One of these values:</td>
</tr>
<tr>
<td></td>
<td>• Processing—Image is being rendered.</td>
</tr>
<tr>
<td></td>
<td>• Failed—Rendering process failed.</td>
</tr>
<tr>
<td></td>
<td>• Success—Rendering process was successful.</td>
</tr>
<tr>
<td></td>
<td>• Na—Rendering is not available for this image.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>title</th>
<th>String</th>
<th>Title of the file.</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>versionNumber</th>
<th>String</th>
<th>File's version number.</th>
</tr>
</thead>
</table>

### ConnectApi.FileAsset

An asset file.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>baseAssetUrl</td>
<td>String</td>
<td>Base download URL of the asset.</td>
<td>45.0</td>
</tr>
<tr>
<td>baseUnauthenticatedAssetUrl</td>
<td>String</td>
<td>Base download URL of the asset for unauthenticated users if isVisibleByExternalUsers is true, otherwise null.</td>
<td>45.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the asset.</td>
<td>45.0</td>
</tr>
<tr>
<td>isVisibleByExternalUsers</td>
<td>Boolean</td>
<td>Indicates whether unauthenticated users can see the asset file (true) or not (false).</td>
<td>45.0</td>
</tr>
<tr>
<td>masterLabel</td>
<td>String</td>
<td>Label of the asset.</td>
<td>45.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Unique name of the asset.</td>
<td>45.0</td>
</tr>
<tr>
<td>namespacePrefix</td>
<td>String</td>
<td>Namespace prefix of the package containing the asset.</td>
<td>45.0</td>
</tr>
</tbody>
</table>
### Property Name | Type | Description | Available Version
--- | --- | --- | ---
type | String | Type of asset. | 45.0

SEE ALSO:
- ConnectApi.Recommendation
- ConnectApi.NBANativeRecommendation

## ConnectApi.FilePreview

A file preview.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
</table>
| format | ConnectApi.FilePreviewFormat | The format of the preview. Values are:  
  - Jpg—Preview format is JPG.  
  - Pdf—Preview format is PDF.  
  - Svg—Preview format is compressed SVG.  
  - Thumbnail—Preview format is 240 x 180 PNG.  
  - ThumbnailBig—Preview format is 720 x 480 PNG.  
  - ThumbnailTiny—Preview format is 120 x 90 PNG. | 39.0 |
| previewUrlCount | Integer | The total number of preview URLs for this preview format. | 39.0 |
| previewUrls | List<ConnectApi.FilePreviewUrl> | A list of file preview URLs. | 39.0 |
| status | ConnectApi.FilePreviewStatus | The availability status of the preview. Values are:  
  - Available—Preview is available.  
  - InProgress—Preview is being processed.  
  - NotAvailable—Preview is unavailable.  
  - NotScheduled—Generation of the preview isn’t scheduled yet. | 39.0 |
| url | String | The URL for the file preview. | 39.0 |

SEE ALSO:
- ConnectApi.FilePreviewCollection

## ConnectApi.FilePreviewCollection

A collection of file previews.
### ConnectApi.FilePreviewUrl

A URL to a file preview.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>pageNumber</td>
<td>Integer</td>
<td>Preview page number starting from zero, or null for PDF files.</td>
<td>39.0</td>
</tr>
<tr>
<td>previewUrl</td>
<td>String</td>
<td>File preview URL.</td>
<td>39.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.FilePreview

### ConnectApi.FilesCapability

If a feed element has this capability, it has one or more file attachments.

Subclass of ConnectApi.FeedElementCapability.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>items</td>
<td>List&lt;ConnectApi.Content&gt;</td>
<td>Collection of files.</td>
<td>36.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.FeedElementCapabilities

### ConnectApi.FileSummary

A file summary.

Subclass of ConnectApi.File.

No additional properties.
ConnectApi.FindRoutesWithFewestSplitsOutputRepresentation

A list of inventory location combinations that can fulfill an order without exceeding the maximum number of shipments.
Subclass of ConnectApi.BaseOutputRepresentation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>targetLocations</td>
<td>List&lt;ConnectApi.AvailableLocationOutputRepresentation&gt;</td>
<td>Each element of the list is a set of inventory locations that together can fulfill the order being routed.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
findRoutesWithFewestSplits(findRoutesWithFewestSplitsInputRepresentation)

ConnectApi.FindRoutesWithFewestSplitsUsingOCIOutputRepresentation

A list of order fulfillment routes with inventory availability information.
Subclass of ConnectApi.BaseOutputRepresentation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>results</td>
<td>List&lt;ConnectApi.FindRoutesWithFewestSplitsWithInventoryOutputRepresentation&gt;</td>
<td>Each element of the list is the response for one element of the input list.</td>
<td>54.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
findRoutesWithFewestSplitsUsingOCI(findRoutesWithFewestSplitsUsingOCIInput)

ConnectApi.FindRoutesWithFewestSplitsWithInventoryOutputRepresentation

Sets of inventory locations that can combine to fulfill an order, with availability data for those locations.
Subclass of ConnectApi.BaseOutputRepresentation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>inventory</td>
<td>ConnectApi.OCIGetInventoryAvailabilityOutputRepresentation</td>
<td>Inventory availability data for the location groups and locations specified in the input.</td>
<td>54.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>targetLocations</td>
<td>List&lt;ConnectApi.AvailableLocation&gt;</td>
<td>Each entry in the list is a set of inventory locations that can combine to fulfill an order.</td>
<td>54.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- findRoutesWithFewestSplitsUsingOCI(findRoutesWithFewestSplitsUsingOCIInput)
- ConnectApi.FindRoutesWithFewestSplitsUsingOCIOutputRepresentation

**ConnectApi.FollowerPage**

Page of followers.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>28.0</td>
</tr>
<tr>
<td>followers</td>
<td>List&lt;ConnectApi.Subscription&gt;</td>
<td>List of subscriptions.</td>
<td>28.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or null if there isn’t a next page.</td>
<td>28.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the previous page, or null if there isn’t a previous page.</td>
<td>28.0</td>
</tr>
<tr>
<td>total</td>
<td>Integer</td>
<td>Total number of followers across all pages.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

**ConnectApi.FollowingCounts**

Following counts.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>people</td>
<td>Integer</td>
<td>Number of people user is following.</td>
<td>28.0</td>
</tr>
<tr>
<td>records</td>
<td>Integer</td>
<td>Number of records user is following. Topics are a type of record that can be followed as of version 29.0.</td>
<td>28.0</td>
</tr>
<tr>
<td>total</td>
<td>Integer</td>
<td>Total number of items user is following.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.UserDetail

**ConnectApi.FollowingPage**

Page of following subscriptions.
### ConnectApi.FollowIntents

A list of follow intents for a social persona.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>follows</td>
<td>List&lt;ConnectApi.FollowSocialPersonaIntent&gt;</td>
<td>List of follow intents for the social persona.</td>
<td>45.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.SocialPostIntents

### ConnectApi.FollowSocialPersonaIntent

Follow intent on a social persona.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>managedSocialAccount</td>
<td>ConnectApi.ManagedSocialAccount</td>
<td>Managed social account that follows the social persona.</td>
<td>45.0</td>
</tr>
<tr>
<td>socialPersonaId</td>
<td>String</td>
<td>ID of the social persona to follow.</td>
<td>45.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.FollowIntents

### ConnectApi.Form

Marketing integration form.
Available Version | Description | Type | Property Name
---|---|---|---
53.0 | ID of the data extension associated with the marketing integration form. | String | dataExtensionId
53.0 | List of form fields associated with the marketing integration form. | ConnectApi.FormFields | formFieldsList
53.0 | ID of the marketing integration form. | String | formId
53.0 | Name of the marketing integration form. | String | formName

### ConnectApi.FormField
Marketing integration form field.

| Property Name | Type | Description | Available Version
---|---|---|---
name | String | Name of the marketing integration form field. | 53.0

*SEE ALSO:* [ConnectApi.FormFields](#)

### ConnectApi.FormFields
List of marketing integration form fields.

| Property Name | Type | Description | Available Version
---|---|---|---
formFields | List<ConnectApi.FormField> | List of form fields associated with the marketing integration form. | 53.0

*SEE ALSO:* [ConnectApi.Form](#)

### ConnectApi.FormSubmission
Marketing integration form submission.
### ConnectApi.FormulaScope

Formula scope for a target.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>contextValues</td>
<td>Map&lt;String, String&gt;</td>
<td>Map of context values for the scope.</td>
<td>50.0–51.0</td>
</tr>
<tr>
<td>contextValuesMap</td>
<td>Map&lt;String, Object&gt;</td>
<td>Map of context values for the scope.</td>
<td>52.0</td>
</tr>
<tr>
<td>fields</td>
<td>List&lt;String&gt;</td>
<td>List of fields of the scope.</td>
<td>50.0</td>
</tr>
<tr>
<td>formula</td>
<td>String</td>
<td>Formula of the scope.</td>
<td>50.0</td>
</tr>
</tbody>
</table>

**Note:** In version 52.0 and later, use the `contextValuesMap` property.

### ConnectApi.FulfillmentGroupOutputRepresentation

Information about one FulfillmentOrder from a request to create fulfillment orders from multiple OrderDeliveryGroupSummaries. If the FulfillmentOrder was created, then its ID is returned. If it failed, then data from the input is returned so you can resubmit it.

Subclass of `ConnectApi.BaseOutputRepresentation`.

| Property Name                          | Type                                                                 | Description                                                                 | Available Version |
|----------------------------------------|                                                                     |                                                                            |                  |
| fulfilledFromLocationId                | String                                                              | (Creation failed) The input FulfilledFromLocationId.                        | 50.0              |
| fulfillmentOrderId                     | String                                                              | The FulfillmentOrderId from the successfully created FulfillmentOrder.     | 50.0              |
| fulfillmentType                        | String                                                              | (Creation failed) The input FulfillmentType.                               | 50.0              |
| orderDeliveryGroupSummaryId            | String                                                              | (Creation failed) The input OrderDeliveryGroupSummaryId.                   | 50.0              |
| orderItemSummaries                     | List<ConnectApi.OrderItemSummaryOutputRepresentation>               | (Creation failed) The input list of OrderItemSummaries.                     | 50.0              |
| orderSummaryId                         | String                                                              | (Creation failed) The input OrderSummaryId.                                | 50.0              |
### ConnectApi.FulfillmentOrderCancelLineItemsOutputRepresentation

Wraps the base output.

Subclass of `ConnectApi.BaseOutputRepresentation`.

No additional properties.

### ConnectApi.FulfillmentOrderInvoiceOutputRepresentation

ID of the created invoice.

Subclass of `ConnectApi.BaseOutputRepresentation`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>invoiceId</td>
<td>String</td>
<td>ID of the created invoice.</td>
<td>48.0</td>
</tr>
</tbody>
</table>

### ConnectApi.FulfillmentOrderOutputRepresentation

A list of IDs of the created FulfillmentOrders.

Subclass of `ConnectApi.BaseOutputRepresentation`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>fulfillment</td>
<td>List&lt;String&gt;</td>
<td>A list of IDs of created Fulfillment Orders.</td>
<td>48.0</td>
</tr>
<tr>
<td>OrderIds</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### ConnectApi.GatewayLogResponse

Gateway log output.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>createdDate</td>
<td>Datetime</td>
<td>Date when the gateway log was created.</td>
<td>50.0</td>
</tr>
<tr>
<td>gatewayResultCode</td>
<td>String</td>
<td>Result codes that show the status of a transaction as it is passed to the financial institution and then returned to the client.</td>
<td>50.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the gateway log record.</td>
<td>50.0</td>
</tr>
<tr>
<td>interactionStatus</td>
<td>String</td>
<td>Gateway interaction status. It can be SUCCESS, FAILED, or TIMEOUT.</td>
<td>50.0</td>
</tr>
</tbody>
</table>
ConnectApi.GenericBundleCapability

If a feed element has this capability, the feed element has a group of other feed elements condensed into one feed element. This group is called a bundle.

Subclass of ConnectApi.BundleCapability.

ConnectApi.GenericFeedElement

A concrete implementation of the abstract ConnectApi.FeedElement class.

Subclass of ConnectApi.FeedElement.

ConnectApi.GetFOCapacityValuesOutputRepresentation

Response to a request for fulfillment order capacity values for one or more locations.

Subclass of ConnectApi.BaseOutputRepresentation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>locations</td>
<td>List&lt;ConnectApi.LocationCapacityOutputRepresentation&gt;</td>
<td>List of fulfillment order capacity values for one or more locations.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

ConnectApi.GlobalInfluence

Chatter influence.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>percentile</td>
<td>String</td>
<td>Percentile value for the user's influence rank within the org or Experience Cloud site.</td>
<td>28.0</td>
</tr>
<tr>
<td>rank</td>
<td>Integer</td>
<td>Number indicating the user's influence rank, relative to all other users within the org or Experience Cloud site.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

ConnectApi.UserDetail

ConnectApi.GroupChatterSettings

A user's Chatter settings for a specific group.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>emailFrequency</td>
<td>ConnectApi.GroupEmailFrequency</td>
<td>The frequency with which a group member receives email from a group.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

1828
**ConnectApi.GroupInformation**

Describes the Information section of the group. If the group is private, this section is visible only to members.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>text</td>
<td>String</td>
<td>The text of the “Information” section of the group.</td>
<td>28.0</td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>The title of the “Information” section of the group.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.ChatterGroupDetail

**ConnectApi.GroupMember**

Member of a group.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>User’s 18-character ID.</td>
<td>28.0</td>
</tr>
<tr>
<td>lastFeed</td>
<td>Datetime</td>
<td>The date and time at which the group member last accessed the group feed.</td>
<td>31.0</td>
</tr>
<tr>
<td>role</td>
<td>ConnectApi.GroupMembership Type Enum</td>
<td>Type of membership the user has with the group.</td>
<td>28.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• GroupOwner</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• GroupManager</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NotAMember</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NotAMemberPrivateRequested</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• StandardMember</td>
<td></td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>Connect REST API URL to this membership.</td>
<td>28.0</td>
</tr>
<tr>
<td>user</td>
<td>ConnectApi.User Summary</td>
<td>Information about the user who is subscribed to this group.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.GroupMemberPage

**ConnectApi.GroupMemberPage**

Page of group members.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>28.0</td>
</tr>
</tbody>
</table>
### ConnectApi.GroupMembership

Request to become a group member.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>createdDate</td>
<td>Datetime</td>
<td>ISO 8601 date string, for example, 2011-02-25T18:24:31.000Z.</td>
<td>28.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID for the group membership request object.</td>
<td>28.0</td>
</tr>
<tr>
<td>lastUpdateDate</td>
<td>Datetime</td>
<td>ISO 8601 date string, for example, 2011-02-25T18:24:31.000Z.</td>
<td>28.0</td>
</tr>
<tr>
<td>requestedGroup</td>
<td>ConnectApi. Reference</td>
<td>Information about the group the context user is requesting to join.</td>
<td>28.0</td>
</tr>
<tr>
<td>responseMessage</td>
<td>String</td>
<td>A message for the user if their membership request is declined. The value of this property is used only when the value of the status property is Declined. The maximum length is 756 characters.</td>
<td>28.0</td>
</tr>
<tr>
<td>status</td>
<td>ConnectApi. GroupMembershipRequestStatusEnum</td>
<td>Status of a request to join a private group. Values are:</td>
<td>28.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Accepted</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Declined</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Pending</td>
<td></td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>URL of the group membership request object.</td>
<td>28.0</td>
</tr>
<tr>
<td>user</td>
<td>ConnectApi.UserSummary</td>
<td>Information about the user requesting membership in a group.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

[ConnectApi.GroupMembershipRequests](#)
ConnectApi.GroupMembershipRequests

Requests to become group members.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>requests</td>
<td>List&lt;ConnectApi.GroupMembershipRequest&gt;</td>
<td>Information about group membership requests.</td>
<td>28.0</td>
</tr>
<tr>
<td>total</td>
<td>Integer</td>
<td>The total number of requests.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

ConnectApi.GroupRecord

A record associated with a group.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>Record’s 18-character ID.</td>
<td>33.0</td>
</tr>
<tr>
<td>record</td>
<td>ConnectApi.ActorWithId</td>
<td>Information about the record associated with the group.</td>
<td>33.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>Record URL.</td>
<td>33.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

ConnectApi.GroupRecordPage

ConnectApi.GroupRecordPage

A paginated list of ConnectApi.GroupRecord objects.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>33.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or null if there isn’t a next page.</td>
<td>33.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the previous page, or null if there isn’t a previous page.</td>
<td>33.0</td>
</tr>
<tr>
<td>records</td>
<td>List&lt;ConnectApi.GroupRecord&gt;</td>
<td>List of records on the current page.</td>
<td>33.0</td>
</tr>
<tr>
<td>totalRecordCount</td>
<td>Integer</td>
<td>Total number of records associated with the group.</td>
<td>33.0</td>
</tr>
</tbody>
</table>
### ConnectApi.HashtagSegment

Hashtag segment.
Subclass of ConnectApi.MessageSegment.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>tag</td>
<td>String</td>
<td>Text of the topic without the hash symbol (#).</td>
<td>28.0</td>
</tr>
<tr>
<td>topicUrl</td>
<td>String</td>
<td>Connect REST API Topics resource that searches for the topic:</td>
<td>28.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/services/data/v59.0/chatter/topics?exactMatch=true&amp;q=topic</td>
<td></td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>Connect REST API Feed Items resource URL that searches for the topic in all feed items in an organization:</td>
<td>28.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/services/data/v59.0/chatter/feed-items?q=topic</td>
<td></td>
</tr>
</tbody>
</table>

### ConnectApi.HideSocialPostIntent

Hide intent for a social post.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isHidden</td>
<td>Boolean</td>
<td>Specifies whether the managed social account hid the social post (true) or not (false).</td>
<td>45.0</td>
</tr>
<tr>
<td>managedSocialAccount</td>
<td>ConnectApi.ManagedSocialAccount</td>
<td>Managed social account that hides the social post.</td>
<td>45.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.SocialPostIntents

### ConnectApi.HoldFOCapacityOutputRepresentation

Response to a request to hold fulfillment order capacity at one or more locations. Can correspond to one action call.
Subclass of ConnectApi.BaseOutputRepresentation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>holdFOCapacityResponses</td>
<td>List&lt;ConnectApi.HoldFOCapacityResponseOutputRepresentation&gt;</td>
<td>List of responses to the requests to hold fulfillment order capacity at one or more locations.</td>
<td>55.0</td>
</tr>
</tbody>
</table>
ConnectApi.HoldFOCapacityResponseOutputRepresentation

Response to a request to hold fulfillment order capacity at one or more locations.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>capacityResponses</td>
<td>List&lt;ConnectApi.CAPACITYRESPONSEOUTPUTREPRESENTATION&gt;</td>
<td>List of responses to the requests to hold fulfillment order capacity at individual locations.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

ConnectApi.Icon

Icon.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>height</td>
<td>Integer</td>
<td>The height of the icon in pixels.</td>
<td>28.0</td>
</tr>
<tr>
<td>width</td>
<td>Integer</td>
<td>The width of the icon in pixels.</td>
<td>28.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>The URL of the icon. This URL is available to unauthenticated users. This URL does not expire.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.CanvasCapability
- ConnectApi.EnhancedLinkCapability
- ConnectApi.SocialPostCapability

ConnectApi.InlineImageSegment

An inline image in the feed body.

Subclass of ConnectApi.MessageSegment.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>altText</td>
<td>String</td>
<td>Alt text for the inline image.</td>
<td>35.0</td>
</tr>
<tr>
<td>contentSize</td>
<td>Integer</td>
<td>Size of the file in bytes.</td>
<td>35.0</td>
</tr>
<tr>
<td>fileExtension</td>
<td>String</td>
<td>Extension of the file, such as gif.</td>
<td>37.0</td>
</tr>
<tr>
<td>thumbnails</td>
<td>ConnectApi.FilePreviewCollection</td>
<td>Information about the available thumbnails for the image.</td>
<td>35.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>URL to the latest version of the inline image.</td>
<td>35.0</td>
</tr>
</tbody>
</table>

ConnectApi.InteractionsCapability

If a feed element has this capability, it has information about user interactions.
Subclass of `ConnectApi.FeedElementCapability`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>count</td>
<td>Long</td>
<td>The number of individual views, likes, and comments on a feed post.</td>
<td>37.0</td>
</tr>
</tbody>
</table>

**Note:** This count appears in the UI under the feed post as the number of views, for example, “5 views.”

---

**SEE ALSO:**
- `ConnectApi.FeedElementCapabilities`
- `ConnectApi.RelatedQuestion`

---

### ConnectApi.Invitation

An invitation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>email</td>
<td>String</td>
<td>Email address of the user.</td>
<td>39.0</td>
</tr>
</tbody>
</table>
| status        | `ConnectApi.GroupViralInvitationsStatus` | Specifies the status of an invitation to join a group. Values are:  
- ActedUponUser—The user was added to the group. An email was sent asking the user to visit the group.
- Invited—An email was sent asking the user to sign up for the org.
- MaxedOutUsers—The group has the maximum allowed members.
- MultipleError—The user wasn’t invited due to multiple errors.
- NoActionNeededUser—The user is already a member of the group.
- NotVisibleToExternalInviter—The user is not accessible to the user sending the invitation.
- Unhandled—The user couldn’t be added to the group for an unknown reason. | 39.0              |
| userId        | String         | ID of the user.                                                             | 39.0              |

**SEE ALSO:**
- `ConnectApi.Invitations`
ConnectApi.Invitations
A collection of invitations.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>invitations</td>
<td>List&lt;ConnectApi.Invitation&gt;</td>
<td>Collection of invitations.</td>
<td>39.0</td>
</tr>
</tbody>
</table>

ConnectApi.KnowledgeArticleVersion
A knowledge article version.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>articleType</td>
<td>String</td>
<td>Type of the knowledge article.</td>
<td>36.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the knowledge article version.</td>
<td>36.0</td>
</tr>
<tr>
<td>knowledgeArticleId</td>
<td>String</td>
<td>ID of the corresponding knowledge article.</td>
<td>36.0</td>
</tr>
<tr>
<td>lastPublishedDate</td>
<td>Datetime</td>
<td>Last published date of the knowledge article.</td>
<td>36.0</td>
</tr>
<tr>
<td>summary</td>
<td>String</td>
<td>Summary of the knowledge article contents.</td>
<td>36.0</td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>Title of the knowledge article.</td>
<td>36.0</td>
</tr>
<tr>
<td>urlName</td>
<td>String</td>
<td>URL name of the knowledge article.</td>
<td>36.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.KnowledgeArticleVersionCollection

ConnectApi.KnowledgeArticleVersionCollection
A collection of knowledge article versions.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>items</td>
<td>List&lt;ConnectApi.KnowledgeArticleVersion&gt;</td>
<td>A collection of knowledge article versions.</td>
<td>36.0</td>
</tr>
</tbody>
</table>

ConnectApi.LabeledRecordField
Record field containing a label and a text value.
This class is abstract.
Subclass of ConnectApi.AbstractRecordField.
Superclass of:
- ConnectApi.CompoundRecordField
- ConnectApi.CurrencyRecordField
- ConnectApi.DateRecordField
ConnectApi.PercentRecordField
ConnectApi.PicklistRecordField
ConnectApi.RecordField
ConnectApi.ReferenceField
ConnectApi.ReferenceWithDateRecordField

**Important:** The composition of a feed can change between releases. Write your code to handle instances of unknown subclasses.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>label</td>
<td>String</td>
<td>Localized string describing the record field.</td>
<td>29.0</td>
</tr>
<tr>
<td>text</td>
<td>String</td>
<td>Text value of the record field. All record fields have a text value. To ensure that all clients can consume new content, inspect the record field’s type property. If it isn’t recognized, render the text value as the default case.</td>
<td>29.0</td>
</tr>
</tbody>
</table>

**ConnectApi.LightningExtensionInformation**

Lightning extension information.
Subclass of ConnectApi.AbstractExtensionInformation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>compositionComponent</td>
<td>String</td>
<td>Component to use in compose state.</td>
<td>40.0</td>
</tr>
<tr>
<td>headerTextLabel</td>
<td>String</td>
<td>Label for the extension’s header.</td>
<td>40.0</td>
</tr>
<tr>
<td>hoverTextLabel</td>
<td>String</td>
<td>Label for hovering over the extension.</td>
<td>40.0</td>
</tr>
<tr>
<td>renderComponent</td>
<td>String</td>
<td>Component to use in render or preview state.</td>
<td>40.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

ConnectApi.ExtensionDefinition

**ConnectApi.LikeIntent**

Like intent for a social post.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isLiked</td>
<td>Boolean</td>
<td>Specifies whether the managed social account liked the social post (true) or not (false).</td>
<td>45.0</td>
</tr>
</tbody>
</table>
### ConnectApi.LikeIntents

List of like intents for a social post.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>likes</td>
<td>List&lt;ConnectApi.LikeIntent&gt;</td>
<td>List of like intents for the social post.</td>
<td>45.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.SocialPostIntents

### ConnectApi.LikeSocialPostIntent

Like intent on a social post.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>socialAccountId</td>
<td>String</td>
<td>ID of the social account that likes the social post.</td>
<td>46.0</td>
</tr>
<tr>
<td>socialPostId</td>
<td>String</td>
<td>ID of the social post to like.</td>
<td>46.0</td>
</tr>
</tbody>
</table>

### ConnectApi.LikeSummary

Summary of a like.
Subclass of ConnectApi.UserFeedEntityActivitySummary.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>likeId</td>
<td>String</td>
<td>ID of the like.</td>
<td>42.0</td>
</tr>
</tbody>
</table>

### ConnectApi.LinkCapability

If a feed element has this capability, it has a link.
Subclass of ConnectApi.FeedElementCapability.
### Available Version
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>String</td>
<td>Link URL. The URL can be to an external site.</td>
<td>32.0</td>
</tr>
<tr>
<td>urlName</td>
<td>String</td>
<td>Description of the link.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- [ConnectApi.FeedElementCapabilities](#)

### ConnectApi.LinkMetadata

Metadata for a link.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the link.</td>
<td>42.0</td>
</tr>
<tr>
<td>frameSource</td>
<td>String</td>
<td>HTML required to display the resource.</td>
<td>42.0</td>
</tr>
<tr>
<td>height</td>
<td>Integer</td>
<td>Height required to display the HTML.</td>
<td>42.0</td>
</tr>
<tr>
<td>originalUrl</td>
<td>String</td>
<td>Original URL that was used to request the metadata.</td>
<td>42.0</td>
</tr>
<tr>
<td>providerUrl</td>
<td>String</td>
<td>URL of the provider that the information is retrieved from.</td>
<td>42.0</td>
</tr>
<tr>
<td>source</td>
<td><a href="#">ConnectApi.LinkMetadataSource</a></td>
<td>Source of the link metadata. Values are:</td>
<td>42.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- None—Link metadata wasn’t retrieved.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- SfDc—Salesforce is the source.</td>
<td></td>
</tr>
<tr>
<td>thumbnailUrl</td>
<td>String</td>
<td>Thumbnail of the resource.</td>
<td>42.0</td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>Title of the link.</td>
<td>42.0</td>
</tr>
<tr>
<td>type</td>
<td><a href="#">ConnectApi.LinkMetadataType</a></td>
<td>Type of link that the metadata represents. Values are:</td>
<td>42.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Error—Link metadata couldn’t be retrieved.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Link—Represents a link.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- None—Link metadata wasn’t retrieved because the link isn’t an allowed domain.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Photo—Represents a photo.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Rich—Represents rich content, typically HTML content.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Unknown—Link metadata was retrieved, but the type is unknown.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Video—Represents a video.</td>
<td></td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>URL of the image to display, if one is available.</td>
<td>42.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------</td>
<td>------------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>width</td>
<td>Integer</td>
<td>Width required to display the HTML.</td>
<td>42.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
*ConnectApi.LinkMetadataCollection*

**ConnectApi.LinkMetadataCollection**

Collection of link metadata.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>linkMetadataList</td>
<td>List</td>
<td>List of metadata for links.</td>
<td>42.0</td>
</tr>
</tbody>
</table>

**ConnectApi.LinkSegment**

Link segment.
Subclass of *ConnectApi.MessageSegment*.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>String</td>
<td>The link URL.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

**ConnectApi.LocationOutputRepresentation**

An inventory location’s distance to an order recipient.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>distance</td>
<td>Double</td>
<td>The distance from the location to the order recipient.</td>
<td>51.0</td>
</tr>
<tr>
<td>locationIdentifier</td>
<td>String</td>
<td>The location identifier.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

**ConnectApi.LocationCapacityOutputRepresentation**

Fulfillment order capacity values for a location.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>assigned</td>
<td>Integer</td>
<td>Value of the location’s Assigned Fulfillment Order Count.</td>
<td>55.0</td>
</tr>
<tr>
<td>capacity</td>
<td>Integer</td>
<td>Value of the location’s Fulfillment Order Capacity. This property represents the location’s maximum capacity.</td>
<td>55.0</td>
</tr>
</tbody>
</table>
### ConnectApi.ErrorResponse

Error returned by the request, if any.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>error</td>
<td>ConnectApi.ErrorResponse</td>
<td></td>
<td>55.0</td>
</tr>
</tbody>
</table>

### ConnectApi.heldCapacity

Number of fulfillment orders that the location is holding capacity for.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>heldCapacity</td>
<td>Integer</td>
<td></td>
<td>55.0</td>
</tr>
</tbody>
</table>

### ConnectApi.locationId

ID of the location.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>locationId</td>
<td>String</td>
<td></td>
<td>55.0</td>
</tr>
</tbody>
</table>

### ConnectApi.MaintenanceInfo

Information about the upcoming scheduled maintenance for the organization.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the maintenance.</td>
<td>34.0</td>
</tr>
<tr>
<td>maintenanceTitle</td>
<td>String</td>
<td>Title of the maintenance.</td>
<td>34.0</td>
</tr>
<tr>
<td>maintenanceType</td>
<td>ConnectApi.MaintenanceType</td>
<td>Type of maintenance. Values are:</td>
<td>34.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Downtime—Downtime maintenance.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• GenerallyAvailable—Maintenance with generally available mode.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• MaintenanceWithDowntime—Scheduled maintenance with downtime.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>•ReadOnly—Maintenance with read-only mode.</td>
<td></td>
</tr>
<tr>
<td>message</td>
<td>Datetime</td>
<td>Effective time when users start seeing the maintenance message.</td>
<td>34.0</td>
</tr>
<tr>
<td>EffectiveTime</td>
<td>Datetime</td>
<td></td>
<td></td>
</tr>
<tr>
<td>message</td>
<td>Datetime</td>
<td>Expiration time of the maintenance message.</td>
<td>34.0</td>
</tr>
<tr>
<td>ExpirationTime</td>
<td>Datetime</td>
<td></td>
<td></td>
</tr>
<tr>
<td>scheduledEnd</td>
<td>Datetime</td>
<td>Scheduled end of downtime. null for GenerallyAvailable and ReadOnly maintenance types.</td>
<td>34.0</td>
</tr>
<tr>
<td>Downtime</td>
<td>Datetime</td>
<td></td>
<td></td>
</tr>
<tr>
<td>scheduledEnd</td>
<td>Datetime</td>
<td>Scheduled end of maintenance. null for Downtime maintenance type.</td>
<td>34.0</td>
</tr>
<tr>
<td>MaintenanceTime</td>
<td>Datetime</td>
<td></td>
<td></td>
</tr>
<tr>
<td>scheduledStart</td>
<td>Datetime</td>
<td>Scheduled start of downtime. null for GenerallyAvailable and ReadOnly maintenance types.</td>
<td>34.0</td>
</tr>
<tr>
<td>Downtime</td>
<td>Datetime</td>
<td></td>
<td></td>
</tr>
<tr>
<td>scheduledStart</td>
<td>Datetime</td>
<td>Scheduled start time of maintenance. null for Downtime maintenance type.</td>
<td>34.0</td>
</tr>
<tr>
<td>MaintenanceTime</td>
<td>Datetime</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**SEE ALSO:**

ConnectApi.OrganizationSettings
ConnectApi.ManagedContentAssociations

Content topics associated with managed content.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>topics</td>
<td>List&lt;ConnectApi.TopicSummary&gt;</td>
<td>A collection of topics associated with the managed content.</td>
<td>47.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

ConnectApi.ManagedContentVersion

ConnectApi.ManagedContentChannel

Managed content channel.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>cacheControlMaxAge</td>
<td>Long</td>
<td>HTTP cache control max age response header for content delivered from the channel.</td>
<td>55.0</td>
</tr>
<tr>
<td>channelId</td>
<td>String</td>
<td>ID of the managed content channel.</td>
<td>48.0</td>
</tr>
<tr>
<td>channelName</td>
<td>String</td>
<td>Name of the managed content channel.</td>
<td>48.0</td>
</tr>
<tr>
<td>channelType</td>
<td>ConnectApi.ManagedContentChannelType</td>
<td>Type of managed content channel. Values are:</td>
<td>48.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• CloudToCloud—Cloud-to-Cloud integrated channel.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Community—Experience Cloud site channel.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ConnectedApp—Channel served by a connected app.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• PublicUnauthenticated—Public channel. All published content is publicly available.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• UserPermission—Channel backed by a system permission. All published content is available only to users with the permission.</td>
<td></td>
</tr>
<tr>
<td>domain</td>
<td>String</td>
<td>ID or name of the domain assigned to the channel.</td>
<td>52.0</td>
</tr>
<tr>
<td>domainId</td>
<td>String</td>
<td>ID of the domain assigned to the channel.</td>
<td>50.0–51.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>In version 52.0 and later, this information is returned in the domain property.</td>
<td></td>
</tr>
<tr>
<td>domainName</td>
<td>String</td>
<td>Name of the domain assigned to the channel.</td>
<td>50.0</td>
</tr>
<tr>
<td>isChannelSearchable</td>
<td>Boolean</td>
<td>Specifies whether the text contents of the channel are searchable (true) or not (false).</td>
<td>48.0</td>
</tr>
</tbody>
</table>
**Available Version** | **Description** | **Type** | **Property Name**
--- | --- | --- | ---
50.0 | Specifies whether the domain is locked and can't be changed (true) or not (false). | Boolean | isDomainLocked
57.0 | HTTP cache control max age response header for media delivered from the channel. | Long | mediaCacheControlMaxAge

SEE ALSO:

ConnectApi.ManagedContentChannelCollection

### ConnectApi.ManagedContentChannelCollection

Collection of managed content channels.

**Property Name** | **Type** | **Description** | **Available Version**
--- | --- | --- | ---
channels | List<ConnectApi.ManagedContentChannel> | List of managed content channels. | 48.0
currentPageUrl | String | Connect REST API URL identifying the current page. | 48.0
nextPageUrl | String | Connect REST API URL identifying the next page, or null if there isn’t a next page. | 48.0
previousPageUrl | String | Connect REST API URL identifying the previous page, or null if there isn’t a previous page. | 48.0
totalChannels | Integer | Total number of managed content channels. | 48.0

### ConnectApi.ManagedContentChannelDetail

Managed content channel detail.

**Property Name** | **Type** | **Description** | **Available Version**
--- | --- | --- | ---
channelId | String | ID of the managed content channel. | 54.0
channelName | String | Name of the managed content channel. | 54.0
channelType | ConnectApi.ManagedContentChannelType | Type of managed content channel. Values are:
- CloudToCloud—Cloud-to-Cloud integrated channel.
- Community—Experience Cloud site channel.
- ConnectedApp—Channel served by a connected app.
- PublicUnauthenticated—Public channel. All published content is publicly available. | 54.0
UserPermission—Channel backed by a system permission. All published content is available only to users with the permission.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>domain</td>
<td>String</td>
<td>Domain assigned to the managed content channel.</td>
<td>54.0</td>
</tr>
<tr>
<td>domainName</td>
<td>String</td>
<td>Name of the domain assigned to the managed content channel.</td>
<td>54.0</td>
</tr>
<tr>
<td>isChannelSearchable</td>
<td>Boolean</td>
<td>Specifies whether the text of the channel’s contents is searchable (true) or not (false).</td>
<td>54.0</td>
</tr>
<tr>
<td>isDomainLocked</td>
<td>Boolean</td>
<td>Specifies whether the channel’s domain is locked and can’t be changed (true) or not (false).</td>
<td>54.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ManagedContentChannelSummary**

Managed content channel.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>domainUrl</td>
<td>String</td>
<td>Domain URL of the channel.</td>
<td>55.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the managed content channel.</td>
<td>54.0</td>
</tr>
<tr>
<td>resourceUrl</td>
<td>String</td>
<td>Resource URL to complete information of the channel.</td>
<td>54.0</td>
</tr>
<tr>
<td>target</td>
<td>ConnectApi.ManagedContentChannelTargetSummary</td>
<td>Target site associated with the channel.</td>
<td>54.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.ManagedContentDeliveryDocument
- ConnectApi.ManagedContentCollectionItems
- ConnectApi.ManagedContentDeliveryDocumentCollection

**ConnectApi.ManagedContentChannelTargetSummary**

Target site associated with the channel.
### ConnectApi.ManagedContentChannelSummary

Managed content channel summary.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the site associated with the channel.</td>
<td>54.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.ManagedContentChannelSummary

### ConnectApi.ManagedContentCollectionItem

Managed content collection item.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>body</td>
<td>Map&lt;String, Object&gt;</td>
<td>Map of properties of the collection item with their values.</td>
<td>56.0</td>
</tr>
<tr>
<td>contentType</td>
<td>ConnectApi.ManagedContentCollectionItemTypeSummary</td>
<td>Type of collection item.</td>
<td>56.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the collection item.</td>
<td>56.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name or title for collection item.</td>
<td>56.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.ManagedContentCollectionItems

### ConnectApi.ManagedContentCollectionItems

Managed content collection Items.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>channelInfo</td>
<td>ConnectApi.ManagedContentChannelSummary</td>
<td>Information about the managed content channel.</td>
<td>56.0</td>
</tr>
<tr>
<td>collectionKey</td>
<td>String</td>
<td>Unique identifier for the collection.</td>
<td>56.0</td>
</tr>
<tr>
<td>collectionType</td>
<td>ConnectApi.ManagedContentTypeSummary</td>
<td>Type of collection.</td>
<td>56.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the collection.</td>
<td>56.0</td>
</tr>
<tr>
<td>items</td>
<td>List&lt;ConnectApi.ManagedContentCollectionItem&gt;</td>
<td>List of collection items.</td>
<td>56.0</td>
</tr>
<tr>
<td>language</td>
<td>String</td>
<td>Language locale of the collection.</td>
<td>56.0</td>
</tr>
</tbody>
</table>
### ConnectApi.ManagedContentCollectionItemTypeSummary

Summary of a collection item type.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>fullyQualifiedName</td>
<td>String</td>
<td>Fully qualified name of the collection item type.</td>
<td>56.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the collection item type.</td>
<td>56.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- [ConnectApi.ManagedContentCollectionItem](#)
- [ConnectApi.ManagedContentDateAndTimeNodeValue](#)
- [ConnectApi.ManagedContentDateNodeValue](#)
- [ConnectApi.ManagedContentDeliveryDocument](#)

### ConnectApi.ManagedContentDateAndTimeNodeValue

Managed content node of date and time type.

Subclass of [ConnectApi.ManagedContentNodeValue](#).

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>dateTimeValue</td>
<td>Datetime</td>
<td>UTC date and time value of the managed content node.</td>
<td>48.0</td>
</tr>
<tr>
<td>timeZone</td>
<td>String</td>
<td>Time zone in which the date and time is authored.</td>
<td>48.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ManagedContentDateNodeValue

Managed content node of date type.

Subclass of [ConnectApi.ManagedContentNodeValue](#).

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>value</td>
<td>Datetime</td>
<td>Date value of the managed content node.</td>
<td>48.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ManagedContentDeliveryDocument

Managed content in delivery scope.
Subclass of ConnectApi.AbstractManagedContentDeliveryDocument in version 55.0 and later. Properties with an available version of 54.0 only are included in ConnectApi.AbstractManagedContentDeliveryDocument in version 55.0 and later.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>channelInfo</td>
<td>ConnectApi.ManagedContentChannelSummary</td>
<td>Information about the managed content channel.</td>
<td>54.0</td>
</tr>
<tr>
<td>contentBody</td>
<td>Map&lt;String, Object&gt;</td>
<td>Map of properties of the managed content with their values.</td>
<td>54.0</td>
</tr>
<tr>
<td>contentKey</td>
<td>String</td>
<td>Globally unique identifier (GUID) for the managed content.</td>
<td>54.0 only</td>
</tr>
<tr>
<td>contentType</td>
<td>ConnectApi.ManagedContentTypeSummary</td>
<td>Type of managed content.</td>
<td>54.0 only</td>
</tr>
<tr>
<td>language</td>
<td>String</td>
<td>Language locale of the managed content.</td>
<td>54.0 only</td>
</tr>
<tr>
<td>managedContentId</td>
<td>String</td>
<td>ID of the managed content.</td>
<td>54.0 only</td>
</tr>
<tr>
<td>publishedDate</td>
<td>Datetime</td>
<td>Most recent publish date of the managed content.</td>
<td>54.0 only</td>
</tr>
<tr>
<td>references</td>
<td>Map&lt;String, ConnectApi.AbstractManagedContentReference&gt;</td>
<td>Map of references with contentKey as the key.</td>
<td>54.0</td>
</tr>
<tr>
<td>referencesList</td>
<td>List&lt;ConnectApi.AbstractManagedContentReference&gt;</td>
<td>List of references.</td>
<td>54.0</td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>Title of the managed content.</td>
<td>54.0 only</td>
</tr>
<tr>
<td>unauthenticatedUrl</td>
<td>String</td>
<td>Public URL for the managed content.</td>
<td>54.0 only</td>
</tr>
<tr>
<td>urlName</td>
<td>String</td>
<td>URL name of the managed content.</td>
<td>54.0 only</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.ManagedContentDeliveryDocumentCollection

**ConnectApi.ManagedContentDeliveryDocumentCollection**

Managed content delivery document collection.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>channelInfo</td>
<td>ConnectApi.ManagedContentChannelSummary</td>
<td>Information about the managed content channel.</td>
<td>55.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>-----------------------</td>
<td>------------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>contents</td>
<td>List&lt;ConnectApi.AbstractManagedContentDeliveryDocument&gt;</td>
<td>List of managed content delivery documents.</td>
<td>55.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>URL to the current page of managed content records.</td>
<td>55.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>URL to the next page of managed content records.</td>
<td>55.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>URL to the previous page of managed content records.</td>
<td>55.0</td>
</tr>
<tr>
<td>references</td>
<td>Map&lt;String, ConnectApi.AbstractManagedContentReference&gt;</td>
<td>Map of references with contentKey as the key.</td>
<td>55.0</td>
</tr>
<tr>
<td>referencesList</td>
<td>List&lt;ConnectApi.AbstractManagedContentReference&gt;</td>
<td>List of references.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ManagedContentDeliveryDocumentSummary**

Managed content delivery document summary.

Subclass of ConnectApi.AbstractManagedContentDeliveryDocument.

No additional properties.

**ConnectApi.ManagedContentMediaNodeValue**

Managed content node of media type.

Subclass of ConnectApi.ManagedContentNodeValue.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>altText</td>
<td>String</td>
<td>Alternative text for the managed content node.</td>
<td>47.0</td>
</tr>
<tr>
<td>altUrl</td>
<td>String</td>
<td>Alternative URL to the managed content node.</td>
<td>47.0–48.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>In version 49.0 and later, this information is returned in the thumbnailUrl property.</td>
<td></td>
</tr>
<tr>
<td>contentKey</td>
<td>String</td>
<td>Content key of the managed content.</td>
<td>51.0</td>
</tr>
<tr>
<td>fileName</td>
<td>String</td>
<td>File name of the managed content node.</td>
<td>48.0</td>
</tr>
<tr>
<td>mediaType</td>
<td>ConnectApi.ManagedContentMediaType</td>
<td>Type of managed content media. Value is Image.</td>
<td>47.0</td>
</tr>
<tr>
<td>mimeType</td>
<td>String</td>
<td>MIME type of the managed content node.</td>
<td>47.0</td>
</tr>
</tbody>
</table>
### ConnectApi.ManagedContentMediaSourceNodeValue

Source of managed content media.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>fileName</td>
<td>String</td>
<td>File name of the media source.</td>
<td>49.0</td>
</tr>
<tr>
<td>isExternal</td>
<td>Boolean</td>
<td>Specifies whether the media source is referenced via an external URL (true) or uploaded (false).</td>
<td>49.0</td>
</tr>
</tbody>
</table>
| mediaType         | ConnectApi.ManagedContentMediaType | Type of managed content media. Values are:  
|                   |                           | • Document  
|                   |                           | • Image                                                       | 49.0              |
| mimeType          | String                    | MIME type of the media source.                                              | 49.0              |
| referenceId       | String                    | Reference ID of the uploaded media source.                                   | 49.0              |
| resourceUrl       | String                    | Resource URL of the media source.                                           | 49.0              |
| unauthenticatedUrl| String                    | URL to the media source for unauthenticated users, or null if the media source isn’t available to external users. | 49.0              |
| url               | String                    | URL to the media source for authenticated users.                             | 49.0              |

### ConnectApi.ManagedContentNodeType

Managed content node type.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>label</td>
<td>String</td>
<td>Label of the managed content node type.</td>
<td>47.0</td>
</tr>
</tbody>
</table>
### ConnectApi.ManagedContentNodeValue

Managed content node.

This class is abstract.

**Superclass of:**
- ConnectApi.ManagedContentDateTimeNodeValue
- ConnectApi.ManagedContentDateNodeValue
- ConnectApi.ManagedContentMediaNodeValue
- ConnectApi.ManagedContentMediaSourceNodeValue
- ConnectApi.ManagedContentTextNodeValue

---

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>Developer name of the managed content node type.</td>
<td>47.0</td>
</tr>
<tr>
<td>nodeType</td>
<td>ConnectApi.ManagedContentNodeTypeEnum</td>
<td>Type of managed content node. Values are:</td>
<td>47.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Date</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- DateTime</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Media</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- MediaSource</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- MultilineText</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- NameField</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- RichText</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Text</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Url</td>
<td></td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.ManagedContentType
### Property Name

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
</table>
| `nodeType`    | `ConnectApi.ManagedContent NodeType` | Type of managed content node. Values are: 
  * Date 
  * DateTime 
  * Media 
  * MediaSource 
  * MultilineText 
  * NameField 
  * RichText 
  * Text 
  * Url | 47.0 |

**SEE ALSO:**

- `ConnectApi.ManagedContentVersion`
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>createdBy</td>
<td>String</td>
<td>ID of the user who created the managed content space.</td>
<td>55.0</td>
</tr>
<tr>
<td>createdDate</td>
<td>Datetime</td>
<td>Date when the managed content space was created.</td>
<td>55.0</td>
</tr>
<tr>
<td>defaultLanguage</td>
<td>String</td>
<td>Default language of the managed content space.</td>
<td>55.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the managed content space.</td>
<td>55.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the managed content space.</td>
<td>55.0</td>
</tr>
<tr>
<td>lastModifiedBy</td>
<td>String</td>
<td>ID of the user who last modified the managed content space.</td>
<td>55.0</td>
</tr>
<tr>
<td>lastModifiedDate</td>
<td>Datetime</td>
<td>Date when the managed content space was last modified.</td>
<td>55.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the managed content space.</td>
<td>55.0</td>
</tr>
<tr>
<td>rootFolderId</td>
<td>String</td>
<td>ID of the root folder of the managed content space.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ManagedContentTextNodeValue**

Managed content node of text type.
Subclass of `ConnectApi.ManagedContentNodeValue`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>value</td>
<td>String</td>
<td>Text value of the managed content node.</td>
<td>47.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ManagedContentType**

Managed content type.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>label</td>
<td>String</td>
<td>Label of the managed content type.</td>
<td>47.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Developer name of the managed content type.</td>
<td>47.0</td>
</tr>
<tr>
<td>nodeTypes</td>
<td>Map&lt;String, ConnectApi.ManagedContentNodeType&gt;</td>
<td>Map of node types for the managed content type.</td>
<td>47.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- `ConnectApi.ManagedContentVersionCollection`
### ConnectApi.ManagedContentTypeSummary
Managed content type.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>fullyQualified</td>
<td>String</td>
<td>Fully qualified name of the managed content type.</td>
<td>54.0</td>
</tr>
<tr>
<td>Name</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Reserved for future use.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.ManagedContentDeliveryDocument
- ConnectApi.ManagedContentCollectionItems

### ConnectApi.ManagedContentVersion
Managed content version.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>associations</td>
<td>ConnectApi.ManagedContent</td>
<td>Content topics associated with the managed content.</td>
<td>47.0</td>
</tr>
<tr>
<td></td>
<td>ManagedContentAssociations</td>
<td></td>
<td></td>
</tr>
<tr>
<td>contentKey</td>
<td>String</td>
<td>Content key of the managed content.</td>
<td>51.0</td>
</tr>
<tr>
<td>contentNodes</td>
<td>Map&lt;String,</td>
<td>Map of content nodes.</td>
<td>47.0</td>
</tr>
<tr>
<td></td>
<td>ConnectApi.ManagedContent</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>NodeValue&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>contentUrlName</td>
<td>String</td>
<td>Content URL name of the managed content version.</td>
<td>48.0</td>
</tr>
<tr>
<td>language</td>
<td>String</td>
<td>Language of the managed content version.</td>
<td>48.0</td>
</tr>
<tr>
<td>managedContentId</td>
<td>String</td>
<td>ID of the managed content.</td>
<td>47.0</td>
</tr>
<tr>
<td>publishedDate</td>
<td>Datetime</td>
<td>Date when the managed content version was last</td>
<td>47.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>published.</td>
<td></td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>Title of the managed content version.</td>
<td>47.0</td>
</tr>
<tr>
<td>type</td>
<td>String</td>
<td>Type of managed content version.</td>
<td>47.0</td>
</tr>
<tr>
<td>typeLabel</td>
<td>String</td>
<td>Type label of the managed content type.</td>
<td>47.0</td>
</tr>
<tr>
<td>unauthenticatedUrl</td>
<td>String</td>
<td>Unauthenticated delivery URL.</td>
<td>50.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.ManagedContentVersionCollection
**ConnectApi.ManagedContentVersionCollection**

Collection of managed content versions.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>47.0</td>
</tr>
<tr>
<td>items</td>
<td>List&lt;ConnectApi.ManagedContentVersion&gt;</td>
<td>List of managed content versions.</td>
<td>47.0</td>
</tr>
<tr>
<td>managedContentTypes</td>
<td>Map&lt;String, ConnectApi.ManagedContentType&gt;</td>
<td>Map of managed content types.</td>
<td>47.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or null if there isn’t a next page.</td>
<td>47.0</td>
</tr>
<tr>
<td>total</td>
<td>Integer</td>
<td>Total number of managed content versions.</td>
<td>47.0</td>
</tr>
<tr>
<td>totalTypes</td>
<td>Integer</td>
<td>Total number of managed content types.</td>
<td>47.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ManagedSocialAccount**

Information describing a managed social account or fan page of a social network.

Subclass of ConnectApi.BaseManagedSocialAccount

No additional properties.

**ConnectApi.ManagedSocialAccounts**

A list of managed social accounts.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>managedSocialAccounts</td>
<td>List&lt;ConnectApi.ManagedSocialAccount&gt;</td>
<td>List of managed social accounts.</td>
<td>44.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ManagedTopic**

Represents a managed topic in an Experience Cloud site.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>children</td>
<td>List&lt;ConnectApi.ManagedTopic&gt;</td>
<td>Children managed topics of the managed topic; null if the depth request parameter isn’t specified or is 1.</td>
<td>35.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of managed topic.</td>
<td>32.0</td>
</tr>
</tbody>
</table>
### ConnectApi.ManagedTopic

Type of managed topic.

- **Content**—Topics that are associated with native content.
- **Featured**—Topics that are featured, for example, on the Experience Cloud site home page, but don’t provide overall navigation.
- **Navigational**—Topics that display in a navigational menu in the Experience Cloud site.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>managedTopicType</td>
<td>ConnectApi.Managed TopicType</td>
<td>Type of managed topic.</td>
<td>32.0</td>
</tr>
<tr>
<td>parent</td>
<td>ConnectApi.Reference</td>
<td>Parent managed topic of the managed topic.</td>
<td>35.0</td>
</tr>
<tr>
<td>topic</td>
<td>ConnectApi.Topic</td>
<td>Information about the topic.</td>
<td>32.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>Connect REST API URL to the managed topic.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

SEE ALSO: [ConnectApi.ManagedTopicCollection](#)

### ConnectApi.ManagedTopicCollection

A collection of managed topics.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>32.0</td>
</tr>
<tr>
<td>managedTopics</td>
<td>List&lt;ConnectApi.ManagedTopic&gt;</td>
<td>List of managed topics.</td>
<td>32.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or <strong>null</strong> if there isn’t a next page.</td>
<td>44.0</td>
</tr>
</tbody>
</table>

### ConnectApiMarkupBeginSegment

The beginning of rich text markup.

Subclass of [ConnectApi.MessageSegment](#).

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>altText</td>
<td>String</td>
<td>Alternative text for the segment, if available.</td>
<td>45.0</td>
</tr>
<tr>
<td>htmlTag</td>
<td>String</td>
<td>The HTML tag for this markup.</td>
<td>35.0</td>
</tr>
<tr>
<td>markupType</td>
<td>ConnectApi. MarkupType</td>
<td>Type of rich text markup.</td>
<td>35.0</td>
</tr>
</tbody>
</table>

- **Bold**—Bold tag.
- **Code**—Code tag.
### ConnectApiMarkupEndSegment

The end of rich text markup.

Subclass of `ConnectApi.MessageSegment`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>url</code></td>
<td><code>String</code></td>
<td>URL to the segment, if available.</td>
<td>45.0</td>
</tr>
</tbody>
</table>

### ConnectApiMediaReference

A media reference.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>mediaUrl</code></td>
<td><code>String</code></td>
<td>URL to stream or download the media.</td>
<td>41.0</td>
</tr>
</tbody>
</table>
### ConnectApi.MediaReferenceCapability

If a feed element has this capability, it has one or more media references.

Subclass of `ConnectApi.FeedElementCapability`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>media</td>
<td>List&lt;ConnectApi.MediaReference&gt;</td>
<td>Collection of media references.</td>
<td>41.0</td>
</tr>
</tbody>
</table>

### ConnectApi.MentionCompletion

Information about a record that could be used to @mention a user or group.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>additionalLabel</td>
<td>String</td>
<td>If one exists, an additional label for the record represented by this completion, for example, “(Customer)” or “(Acme Corporation)”.</td>
<td>29.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>A description of the record represented by this completion.</td>
<td>29.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>The name of the record represented by this completion. The name is localized, if possible.</td>
<td>29.0</td>
</tr>
<tr>
<td>outOfOffice</td>
<td>ConnectApi.OutOfOffice</td>
<td>If the record represented by this completion is a user, an additional out-of-office message, if one exists, for the user.</td>
<td>40.0</td>
</tr>
<tr>
<td>photoUrl</td>
<td>String</td>
<td>A URL to the photo or icon of the record represented by this completion.</td>
<td>29.0</td>
</tr>
<tr>
<td>recordId</td>
<td>String</td>
<td>The ID of the record represented by this completion.</td>
<td>29.0</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>-------------</td>
<td>---------------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>userType</td>
<td>ConnectApi.UserTypeEnum</td>
<td>If the record represented by this completion is a user, this value is the user type associated</td>
<td>30.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>with that user; otherwise the value is <strong>null</strong>.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>One of these values:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ChatterGuest—User is an external user in a private group.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ChatterOnly—User is a Chatter Free customer.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Guest—User is unauthenticated.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Internal—User is a standard org member.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Portal—User is an external user in an Experience Cloud site.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• System—User is Chatter Expert or a system user.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Undefined—User is a user type that is a custom object.</td>
<td></td>
</tr>
</tbody>
</table>

**SEE ALSO:**

[ConnectApi.MentionCompletionPage](#)

**ConnectApi.MentionCompletionPage**

Paginated list of Mention Completion response bodies.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>29.0</td>
</tr>
<tr>
<td>mentionCompletions</td>
<td>List&lt;ConnectApi.</td>
<td>A list of mention completion proposals. Use these proposals to build a feed post body.</td>
<td>29.0</td>
</tr>
<tr>
<td></td>
<td>MentionCompletion&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or <strong>null</strong> if there isn’t a next page.</td>
<td>29.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the previous page, or <strong>null</strong> if there isn’t a previous page.</td>
<td>29.0</td>
</tr>
</tbody>
</table>

**ConnectApi.MentionSegment**

Mention segment.

Subclass of [ConnectApi.MessageSegment](#).

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>accessible</td>
<td>Boolean</td>
<td>Specifies whether the mentioned user or group can see the post in which they are mentioned (true) or not (false).</td>
<td>28.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the mentioned user or group.</td>
<td>28.0</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>record</td>
<td>ConnectApi.ActorWithId</td>
<td>Information about the mentioned user or group.</td>
<td>29.0</td>
</tr>
<tr>
<td>user</td>
<td>ConnectApi.User Summary</td>
<td>Information about the mentioned user.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Important:</strong> In versions 29.0 and later, use the record property.</td>
<td></td>
</tr>
</tbody>
</table>

ConnectApi.MentionValidation

Information about whether a proposed mention is valid for the context user.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>recordId</td>
<td>String</td>
<td>The ID of the mentioned record.</td>
<td>29.0</td>
</tr>
<tr>
<td>validationStatus</td>
<td>ConnectApi.MentionValidation Status Enum</td>
<td>Type of validation error for a proposed mention, if any.</td>
<td>29.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Disallowed—The proposed mention is invalid and is rejected because the context user is trying to mention something that is not allowed. For example, a user who is not a member of a private group is trying to mention the private group.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Inaccessible—The proposed mention is allowed, but the user or record being mentioned isn’t notified. They don’t have access to the parent record that’s being discussed.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Ok—There is no validation error for this proposed mention.</td>
<td></td>
</tr>
</tbody>
</table>

SEE ALSO: ConnectApi.MentionValidations

ConnectApi.MentionValidations

Information about whether a set of mentions is valid for the context user.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>hasErrors</td>
<td>Boolean</td>
<td>Indicates whether at least one of the proposed mentions has an error (true), or not (false). For example, context users can’t mention private groups</td>
<td>29.0</td>
</tr>
</tbody>
</table>
they don’t belong to. If such a group is included in the list of mention validations, hasErrors is true and the group has a validationStatus of Disallowed in its mention validation.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>mentionValidations</td>
<td>List&lt;ConnectApi.MentionValidation&gt;</td>
<td>List of mention validation information in the same order as the provided record IDs.</td>
<td>29.0</td>
</tr>
</tbody>
</table>

**ConnectApi.MessageBody**

Message body.
No additional properties.

SEE ALSO:
- ConnectApi.ChatterLikesCapability
- ConnectApi.ChatterMessage
- ConnectApi.Comment
- ConnectApi.FeedElement
- ConnectApi.FeedItemSummary

**ConnectApi.MessageSegment**

Message segment.
This class is abstract.
Superclass of:
- ConnectApi.ComplexSegment
- ConnectApi.EntityLinkSegment
- ConnectApi.FieldChangeSegment
- ConnectApi.FieldChangeNameSegment
- ConnectApi.FieldChangeValueSegment
- ConnectApi.HashtagSegment
- ConnectApi.InlineImageSegment
- ConnectApi.LinkSegment
- ConnectApi.MarkupBeginSegment
- ConnectApi.MarkupEndSegment
- ConnectApi.MoreChangesSegment
- ConnectApi.ResourceLinkSegment
- ConnectApi.TextSegment
Message segments in a feed item are typed as `ConnectApi.MessageSegment`. Feed item capabilities are typed as `ConnectApi.FeedItemCapability`. Record fields are typed as `ConnectApi.AbstractRecordField`. These classes are all abstract and have several concrete subclasses. At runtime you can use `instanceof` to check the concrete types of these objects and then safely proceed with the corresponding downcast. When you downcast, you must have a default case that handles unknown subclasses.

**Important:** The composition of a feed can change between releases. Write your code to handle instances of unknown subclasses.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>text</td>
<td><code>String</code></td>
<td>Text-only rendition of this segment. If a client encounters an unknown message segment type, it can render this value.</td>
<td>28.0</td>
</tr>
<tr>
<td>type</td>
<td><code>ConnectApi.MessageSegment.Type Enum</code></td>
<td>The message segment type. One of these values:</td>
<td>28.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• EntityLink</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FieldChange</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FieldChangeName</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FieldChangeValue</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Hashtag</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• InlineImage</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Link</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• MarkupBegin</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• MarkupEnd</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Mention</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• MoreChanges</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ResourceLink</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Text</td>
<td></td>
</tr>
</tbody>
</table>

**SEE ALSO:**

`ConnectApi.AbstractMessageBody`

### `ConnectApi.ModerationCapability`

If a feed element has this capability, users in an Experience Cloud site can flag it for moderation.

Subclass of `ConnectApi.FeedElementCapability`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>moderationFlags</td>
<td><code>ConnectApi.ModerationFlags</code></td>
<td>The moderation flags for this feed element. Moderators can view and take action on flagged items.</td>
<td>31.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

`ConnectApi.FeedElementCapabilities`
ConnectApi.ModerationFlagItemDetail

Flag details on a feed item, comment, or file.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>createdBy</td>
<td>String</td>
<td>ID of the user who flagged the item.</td>
<td>40.0</td>
</tr>
<tr>
<td>createdDate</td>
<td>Datetime</td>
<td>Date when the item was flagged.</td>
<td>40.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the moderation flag.</td>
<td>40.0</td>
</tr>
<tr>
<td>moderationType</td>
<td>ConnectApi.</td>
<td>Type of moderation flag. Values are:</td>
<td>40.0</td>
</tr>
<tr>
<td>CommunityFlagType</td>
<td></td>
<td>• FlagAsInappropriate—Flag for inappropriate content.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FlagAsSpam—Flag for spam.</td>
<td></td>
</tr>
<tr>
<td>note</td>
<td>String</td>
<td>Note from user who flagged the item.</td>
<td>40.0</td>
</tr>
<tr>
<td>visibility</td>
<td>ConnectApi.</td>
<td>Visibility behavior of a flag for various user types. Values are:</td>
<td>40.0</td>
</tr>
<tr>
<td>CommunityFlagVisibility</td>
<td></td>
<td>• ModeratorsOnly—The flag is visible only to users with moderation permissions on the flagged element or item.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• SelfAndModerators—The flag is visible to the creator of the flag and to users with moderation permissions on the flagged element or item.</td>
<td></td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.ModerationFlagsCollection

ConnectApi.ModerationFlags

Information about the moderation flags on a feed item, comment, or file.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>flagCount</td>
<td>Integer</td>
<td>Number of moderation flags on this feed item, comment, or file. If the context user is not a moderator, the property is null.</td>
<td>29.0</td>
</tr>
<tr>
<td>flagCountByReason</td>
<td>Map&lt;ConnectApi. CommunityFlagReasonType, Integer&gt;</td>
<td>Number of moderation flags categorized by reason. Values for ConnectApi.CommunityFlagReasonType are:</td>
<td>40.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FlaggedByRule—Moderation rule flagged the item.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FlaggedBySystem—Einstein flagged the item.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FlaggedByUserAsInappropriate—User flagged the item as inappropriate.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FlaggedByUserAsSpam—User flagged the item as spam.</td>
<td></td>
</tr>
</tbody>
</table>
flaggedByMe

**Boolean**

true if the context user flagged the feed item, comment, or file for moderation; false otherwise.

flaggedByMe

**Boolean**

Collection of flags.

flaggedByMe

SEE ALSO:

- ConnectApi.Comment
- ConnectApi.File
- ConnectApi.ModerationCapability

**ConnectApi.ModerationFlagsCollection**

A collection of flags on a feed item, comment, or file.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageToken</td>
<td>String</td>
<td>Token identifying the current page.</td>
<td>40.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>40.0</td>
</tr>
<tr>
<td>flags</td>
<td>List&lt;ConnectApi.ModerationFlag ItemDetail&gt;</td>
<td>List of flag details.</td>
<td>40.0</td>
</tr>
<tr>
<td>nextPageToken</td>
<td>String</td>
<td>Token identifying the next page, or null if there isn’t a next page.</td>
<td>40.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or null if there isn’t a next page.</td>
<td>40.0</td>
</tr>
<tr>
<td>pageSize</td>
<td>Integer</td>
<td>Number of items per page.</td>
<td>40.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

- ConnectApi.ModerationFlags

**ConnectApi.MoreChangesSegment**

In feed items with a large number of tracked changes, the message is formatted as: “changed A, B, and made X more changes.” The MoreChangesSegment contains the “X more changes.”

Subclass of ConnectApi.MessageSegment.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>moreChanges</td>
<td>List&lt;ConnectApi.FieldChange Segment&gt;</td>
<td>Complete list of tracked changes.</td>
<td>29.0</td>
</tr>
</tbody>
</table>
### ConnectApi.Motif

The motif properties contain URLs for small, medium, and large icons that indicate the Salesforce record type. Common record types are files, users, and groups, but all record types have a set of motif icons. Custom object records use their tab style icon. All icons are available to unauthenticated users so that, for example, you can display the motif icons in an email. The motif can also contain the record type’s base color.

The motif images are icons, not user uploaded images or photos. For example, every user has the same set of motif icons.

Custom object records use their tab style icon, for example, the following custom object uses the “boat” tab style:

```json
"motif": {
  "color": "8C004C",
  "largeIconUrl": "/img/icon/custom51_100/boat64.png",
  "mediumIconUrl": "/img/icon/custom51_100/boat32.png",
  "smallIconUrl": "/img/icon/custom51_100/boat16.png",
  "svgIconUrl": null
},
```

Users use the following icons:

```json
"motif": {
  "color": "1797C0",
  "largeIconUrl": "/img/icon/profile64.png",
  "mediumIconUrl": "/img/icon/profile32.png",
  "smallIconUrl": "/img/icon/profile16.png",
  "svgIconUrl": null
},
```

Groups use the following icons:

```json
"motif": {
  "color": "1797C0",
  "largeIconUrl": "/img/icon/groups64.png",
  "mediumIconUrl": "/img/icon/groups32.png",
  "smallIconUrl": "/img/icon/groups16.png",
  "svgIconUrl": null
},
```

Files use the following icons:

```json
"motif": {
  "color": "1797C0",
  "largeIconUrl": "/img/content/content64.png",
  "mediumIconUrl": "/img/content/content32.png",
  "smallIconUrl": "/img/icon/files16.png",
  "svgIconUrl": null
},
```

**Note:** To view the icons in the previous examples, preface the URL with https://instance_name. For example, https://instance_name/img/icon/profile64.png.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>moreChangesCount</td>
<td>Integer</td>
<td>Number of additional changes.</td>
<td>28.0</td>
</tr>
</tbody>
</table>
### ConnectApi.MultipleAsyncOutputRepresentation

IDs of the asynchronous background operations. This output only includes the operation IDs, regardless of whether calls are made to an external payment gateway. It doesn’t include any errors from the operations.

Subclass of `ConnectApi.BaseOutputRepresentation`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>asyncOutputs</td>
<td>List&lt;ConnectApi.AsyncOutputRepresentation&gt;</td>
<td>List of IDs of background operations.</td>
<td>56.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

`multipleEnsureFundsAsync(multipleEnsureFundsInput)`

### ConnectApi.MultipleFulfillmentOrderInvoicesOutputRepresentation

IDs of the created Invoices.

Subclass of `ConnectApi.BaseOutputRepresentation`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>invoiceIds</td>
<td>List&lt;String&gt;</td>
<td>List of IDs of the created Invoices.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

### ConnectApi.MultipleFulfillmentOrderOutputRepresentation

List of responses for the individual FulfillmentOrder creation attempts from a create multiple fulfillment orders request.

Subclass of `ConnectApi.BaseOutputRepresentation`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>fulfillmentOrders</td>
<td>List&lt;ConnectApi.FulfillmentGroupOutputRepresentation&gt;</td>
<td>A list of response data for created and failed FulfillmentOrders.</td>
<td>50.0</td>
</tr>
</tbody>
</table>
ConnectApi.MuteCapability

If a feed element has this capability, users can mute it. Muted feed elements are visible in the muted feed, and invisible in all other feeds that respect mute.

Subclass of ConnectApi.FeedElementCapability.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isMutedByMe</td>
<td>Boolean</td>
<td>Indicates whether the context user muted the feed element.</td>
<td>35.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.FeedElementCapabilities

ConnectApi.MuteSummary

Summary of a mute.

Subclass of ConnectApi.UserFeedEntityActivitySummary.

No additional properties.

ConnectApi.NamedCredential

Named credential associated with an external credential.

ℹ️ **Important**: Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>calloutOptions</td>
<td>ConnectApi. NamedCredential.CalloutOptions</td>
<td>Callout options for the named credential.</td>
<td>58.0</td>
</tr>
<tr>
<td>calloutStatus</td>
<td>ConnectApi. CalloutStatus</td>
<td>Indicates whether a named credential is enabled for callout. Values are:</td>
<td>59.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Disabled</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Enabled</td>
<td></td>
</tr>
<tr>
<td>calloutUrl</td>
<td>String</td>
<td>URL of the named credential in a callout.</td>
<td>58.0</td>
</tr>
<tr>
<td>createdByNamespace</td>
<td>String</td>
<td>Namespace of the package that created the named credential.</td>
<td>59.0</td>
</tr>
<tr>
<td>customHeaders</td>
<td>List&lt;ConnectApi.CredentialCustomHeader&gt;</td>
<td>Custom HTTP headers for the named credential.</td>
<td>58.0</td>
</tr>
<tr>
<td>developerName</td>
<td>String</td>
<td>Fully qualified developer name of the named credential.</td>
<td>56.0</td>
</tr>
</tbody>
</table>
### External credentials used by the named credential

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>Named credential ID.</td>
<td>58.0</td>
</tr>
<tr>
<td>masterLabel</td>
<td>String</td>
<td>Named credential label.</td>
<td>56.0</td>
</tr>
<tr>
<td>networkConnection</td>
<td>ConnectApi. NetworkConnection</td>
<td>PrivateConnect outbound network connection for the named credential.</td>
<td>58.0</td>
</tr>
<tr>
<td>parameters</td>
<td>List&lt;ConnectApi. NamedCredential Parameter&gt;</td>
<td>Named credential parameters.</td>
<td>58.0</td>
</tr>
<tr>
<td>type</td>
<td>ConnectApi. NamedCredentialType</td>
<td>Type of named credential. Values are: PrivateEndpoint, SecuredEndpoint</td>
<td>58.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>Connect REST API URL of the named credential.</td>
<td>58.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.ExternalCredential
- ConnectApi.NamedCredentialList

### Named credential callout options

Named credential callout options.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>allowMergeFields InBody</td>
<td>Boolean</td>
<td>Specifies whether to allow merge fields in the HTTP body (true) or not (false).</td>
<td>58.0</td>
</tr>
<tr>
<td>allowMergeFields InHeader</td>
<td>Boolean</td>
<td>Specifies whether to allow merge fields in the HTTP header (true) or not (false).</td>
<td>58.0</td>
</tr>
<tr>
<td>generate AuthorizationHeader</td>
<td>Boolean</td>
<td>Specifies whether to generate an authorization header (true) or not (false).</td>
<td>58.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.NamedCredential

### List of named credentials

List of named credentials.
### ConnectApi.NamedCredential

Named credential parameter.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>namedCredentials</td>
<td>List&lt;ConnectApi.NamedCredential&gt;</td>
<td>List of named credentials.</td>
<td>58.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the parameter.</td>
<td>58.0</td>
</tr>
<tr>
<td>parameterName</td>
<td>String</td>
<td>Name of the parameter.</td>
<td>58.0</td>
</tr>
<tr>
<td>parameterType</td>
<td>ConnectApi.NamedCredential.ParameterType</td>
<td>Type of named credential parameter. Values are:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- AllowedManagedPackageNamespaces</td>
<td>58.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- ClientCertificate</td>
<td></td>
</tr>
<tr>
<td>parameterValue</td>
<td>String</td>
<td>Value of the parameter.</td>
<td>58.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

- [ConnectApi.NamedCredential](#)

### ConnectApi.NavigationMenuItem

Navigation menu item.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>actionType</td>
<td>ConnectApi.NavigationMenuItem.ActionType</td>
<td>Event, URL type, or modal navigation menu item. Values are:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Event—Event-based navigation.</td>
<td>52.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Note: Event is internal only and can’t be used in custom components.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- ExternalLink—URL outside of your Experience Cloud site.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- InternalLink—Relative URL inside your Experience Cloud site.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Modal—Modal, such as Account Switcher.</td>
<td></td>
</tr>
<tr>
<td>actionValue</td>
<td>String</td>
<td>For Event action type, the event fully qualified name for the navigation menu item. For ExternalLink and InternalLink action types, the route URL for the navigation menu item.</td>
<td>52.0</td>
</tr>
</tbody>
</table>
### ConnectApi.OutputClasses

#### ConnectApi.NavigationMenuItem

**Fully qualified name for the navigation menu item.**

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>imageUrl</td>
<td>String</td>
<td>URL to the image of the navigation menu item.</td>
<td>52.0</td>
</tr>
<tr>
<td>label</td>
<td>String</td>
<td>Label for the navigation menu item.</td>
<td>52.0</td>
</tr>
<tr>
<td>pageReference</td>
<td>ConnectApi.NavigationMenuPageReference</td>
<td>Page reference for the navigation menu item. Page reference is returned only for the Storefront Categories data source.</td>
<td>59.0</td>
</tr>
<tr>
<td>subMenu</td>
<td>List&lt;ConnectApi.NavigationMenuItem&gt;</td>
<td>Submenu for the navigation menu item.</td>
<td>52.0</td>
</tr>
</tbody>
</table>
| target        | ConnectApi.NavigationMenuItemOpenTarget | Target for the navigation menu item. Values are:  
- CurrentWindow—Navigation menu item opens in the current window.  
- NewWindow—Navigation menu item opens in a new window. | 52.0              |

**SEE ALSO:**

- ConnectApi.NavigationMenuItemCollection

#### ConnectApi.NavigationMenuItemCollection

Collection of navigation menu items.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>menuItems</td>
<td>List&lt;ConnectApi.NavigationMenuItem&gt;</td>
<td>Collection of navigation menu items.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

#### ConnectApi.NavigationMenuPageReference

Navigation menu item page reference.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>attributes</td>
<td>Map&lt;String, String&gt;</td>
<td>Attributes for the navigation menu item page reference.</td>
<td>59.0</td>
</tr>
<tr>
<td>state</td>
<td>Map&lt;String, String&gt;</td>
<td>State for the navigation menu item page reference.</td>
<td>59.0</td>
</tr>
<tr>
<td>type</td>
<td>String</td>
<td>Type for the navigation menu item page reference.</td>
<td>59.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

- ConnectApi.NavigationMenuItem

1868
ConnectApi.NBAActionParameter

A parameter for an action.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the parameter.</td>
<td>45.0</td>
</tr>
<tr>
<td>type</td>
<td>String</td>
<td>Type of the parameter.</td>
<td>45.0</td>
</tr>
<tr>
<td>value</td>
<td>String</td>
<td>Value of the parameter.</td>
<td>45.0</td>
</tr>
</tbody>
</table>

ConnectApi.NBAFlowAction

A recommended flow.
Subclass of ConnectApi.AbstractNBAAction.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>flowLabel</td>
<td>String</td>
<td>Label of the recommended flow.</td>
<td>47.0</td>
</tr>
<tr>
<td>flowType</td>
<td>ConnectApi.NBAFlowType</td>
<td>Type of recommended flow. Values are:</td>
<td>47.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AutoLaunchedFlow—Autolaunched flow that runs in the background.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Flow—Screen flow that accepts user inputs.</td>
<td></td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the flow.</td>
<td>45.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the flow.</td>
<td>45.0</td>
</tr>
</tbody>
</table>

ConnectApi.NBANativeRecommendation

A record the user is recommended to take action on.
Subclass of ConnectApi.AbstractNBATarget.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the recommendation.</td>
<td>45.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the recommendation.</td>
<td>45.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>URL to the recommendation.</td>
<td>45.0</td>
</tr>
</tbody>
</table>

ConnectApi.NBARecommendation

A recommendation returned by a recommendation strategy.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>aiModel</td>
<td>String</td>
<td>Reserved for future use.</td>
<td>47.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>-----------------------</td>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>acceptanceLabel</td>
<td>String</td>
<td>Text indicating user acceptance of the recommendation.</td>
<td>45.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the recommendation.</td>
<td>45.0</td>
</tr>
<tr>
<td>externalId</td>
<td>String</td>
<td>External ID of the recommendation. This ID doesn’t need to be a Salesforce 18-character ID. For example, it can be a product number from an external system.</td>
<td>46.0</td>
</tr>
<tr>
<td>imageUrl</td>
<td>String</td>
<td>URL to the asset file to display.</td>
<td>45.0</td>
</tr>
<tr>
<td>recommendationMode</td>
<td>String</td>
<td>Reserved for future use.</td>
<td>46.0</td>
</tr>
<tr>
<td>recommendationScore</td>
<td>Double</td>
<td>Reserved for future use.</td>
<td>46.0</td>
</tr>
<tr>
<td>rejectionLabel</td>
<td>String</td>
<td>Text indicating user rejection of the recommendation.</td>
<td>45.0</td>
</tr>
<tr>
<td>target</td>
<td>ConnectApi.AbstractNBAAction</td>
<td>Target to act on.</td>
<td>45.0</td>
</tr>
<tr>
<td>targetAction</td>
<td>ConnectApi.AbstractNBAAction</td>
<td>Action to recommend.</td>
<td>45.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.NBARecommendations

ConnectApi.NBARecommendations
Recommendations returned by a recommendation strategy.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>debug</td>
<td>String</td>
<td>Runtime debug information recorded during recommendation strategy execution.</td>
<td>45.0</td>
</tr>
<tr>
<td>errors</td>
<td>String</td>
<td>Runtime errors that occurred during recommendation strategy execution.</td>
<td>45.0</td>
</tr>
<tr>
<td>executionId</td>
<td>String</td>
<td>ID of the recommendation strategy execution.</td>
<td>45.0</td>
</tr>
<tr>
<td>onBehalfOfId</td>
<td>String</td>
<td>ID of the user or entity for which the recommendation strategy was executed.</td>
<td>45.0</td>
</tr>
<tr>
<td>recommendations</td>
<td>List&lt;ConnectApi.NBARecommendation&gt;</td>
<td>List of recommendations returned by a recommendation strategy.</td>
<td>45.0</td>
</tr>
<tr>
<td>trace</td>
<td>ConnectApi.StrategyTrace</td>
<td>Trace information for the recommendation strategy execution, if requested.</td>
<td>45.0</td>
</tr>
</tbody>
</table>
ConnectApi.NetworkConnection

External network connection.

**Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>developerName</td>
<td>String</td>
<td>Name of the network connection.</td>
<td>58.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the network connection.</td>
<td>58.0</td>
</tr>
<tr>
<td>masterLabel</td>
<td>String</td>
<td>Label of the network connection.</td>
<td>58.0</td>
</tr>
<tr>
<td>namespacePrefix</td>
<td>String</td>
<td>Namespace prefix of the network connection.</td>
<td>58.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

ConnectApi.NamedCredential

ConnectApi.NewUserAudienceCriteria

Criteria for the new members type of custom recommendation audience.

Subclass of ConnectApi.AudienceCriteria.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>maxDaysInCommunity</td>
<td>Double</td>
<td>The maximum number of days since a user became a site member.</td>
<td>36.0</td>
</tr>
</tbody>
</table>

ConnectApi.OAuthCredentialAuthUrl

OAuth authentication URL for a credential.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>authenticationUrl</td>
<td>String</td>
<td>Authentication URL for the user external credential. Authentication URLs have encoded and escaped special characters. Before using the URL, undo the encoded and escaped characters.</td>
<td>56.0</td>
</tr>
<tr>
<td>externalCredential</td>
<td>String</td>
<td>Fully qualified developer name of the external credential.</td>
<td>56.0</td>
</tr>
<tr>
<td>principalName</td>
<td>String</td>
<td>Name of the external credential named principal.</td>
<td>56.0</td>
</tr>
</tbody>
</table>
| principalType                  | ConnectApi.CredentialPrincipal | Type of credential principal. Values are:  
  • AwsStsPrincipal
  • NamedPrincipal
  • PerUserPrincipal           | 56.0              |
ConnectApi.OauthProviderInfo

OAuth provider information.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>authorizationUrl</td>
<td>String</td>
<td>The URL used for authorization.</td>
<td>37.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>The name of the OAuth service provider.</td>
<td>37.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
  ConnectApi.UserOauthInfo

ConnectApi.OCIBaseOutputRepresentation

Base Omnichannel Inventory output class.

This class is abstract.

Superclass of:
  • ConnectApi.OCIGetInventoryAvailabilityOutputRepresentation
  • ConnectApi.OCIPublishLocationStructureOutputRepresentation
  • ConnectApi.OCIPublishLocationStructureStatusOutputRepresentation
  • ConnectApi.OCIUploadInventoryAvailabilityOutputRepresentation
  • ConnectApi.OCIUploadInventoryAvailabilityStatusOutputRepresentation

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>errors</td>
<td>List&lt;ConnectApi.ErrorResponse&gt;</td>
<td>Any errors that were returned.</td>
<td>51.0</td>
</tr>
<tr>
<td>success</td>
<td>Boolean</td>
<td>Indicates whether the request was successful.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

ConnectApi.OCICreateReservationErrorOutputRepresentation

Error returned from an attempt to create an Omnichannel Inventory reservation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>errorCode</td>
<td>String</td>
<td>The error code.</td>
<td>51.0</td>
</tr>
<tr>
<td>message</td>
<td>String</td>
<td>Details of the error, if available.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

ConnectApi.OCICreateReservationOutputRepresentation

Result of an Omnichannel Inventory reservation creation request.
### ConnectApi.OCICreateReservationSingleOutputRepresentation

Details of an inventory reservation for one product.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>errorCode</td>
<td>String</td>
<td>The error code, if any.</td>
<td>51.0</td>
</tr>
<tr>
<td>locationGroupIdentifier</td>
<td>String</td>
<td>Identifier of the location group where the inventory is reserved.</td>
<td>51.0</td>
</tr>
<tr>
<td>locationIdentifier</td>
<td>String</td>
<td>Identifier of the location where the inventory is reserved.</td>
<td>51.0</td>
</tr>
<tr>
<td>quantity</td>
<td>Double</td>
<td>The reserved quantity of the product.</td>
<td>51.0</td>
</tr>
<tr>
<td>stockKeepingUnit</td>
<td>String</td>
<td>The SKU of the reserved product.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

### ConnectApi.OCIConsumeReservationErrorOutputRepresentation

Response to a request to consume one inventory reservation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>details</td>
<td>ConnectApi.OCIConsumeReservationSingleOutputRepresentation</td>
<td>Details of the consumed reservation, if successful.</td>
<td>51.0</td>
</tr>
<tr>
<td>errorCode</td>
<td>String</td>
<td>Error code, if any.</td>
<td>51.0</td>
</tr>
<tr>
<td>message</td>
<td>String</td>
<td>Details of the error, if available.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

### ConnectApi.OCIConsumeReservationOutputRepresentation

Response to a request to consume one or more inventory reservations.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>details</td>
<td>ConnectApi.OCIConsumeReservationSingleOutputRepresentation</td>
<td>Details of the consumed reservations, if successful.</td>
<td>51.0</td>
</tr>
<tr>
<td>errorCode</td>
<td>String</td>
<td>Error code, if any.</td>
<td>51.0</td>
</tr>
<tr>
<td>message</td>
<td>String</td>
<td>Details of the error, if available.</td>
<td>51.0</td>
</tr>
</tbody>
</table>
### ConnectApi.OCIFulfillReservationSingleOutputRepresentation

Details of a single fulfilled reservation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>actionRequestId</td>
<td>String</td>
<td>The UUID that identifies the original fulfill reservation request.</td>
<td>5.1.0</td>
</tr>
<tr>
<td>externalRefId</td>
<td>String</td>
<td>The external reference ID of the location that fulfilled the reservation.</td>
<td>5.1.0</td>
</tr>
<tr>
<td>locationIdentifier</td>
<td>String</td>
<td>The identifier of the location that fulfilled the reservation.</td>
<td>5.1.0</td>
</tr>
<tr>
<td>quantity</td>
<td>Double</td>
<td>The fulfilled quantity.</td>
<td>5.1.0</td>
</tr>
<tr>
<td>stockKeepingUnit</td>
<td>String</td>
<td>The SKU of the fulfilled product.</td>
<td>5.1.0</td>
</tr>
</tbody>
</table>

### ConnectApi.OCIFutureInventoryOutputRepresentation

An expected future inventory restock.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>expectedDate</td>
<td>Datetime</td>
<td>Date when the future inventory is expected.</td>
<td>5.1.0</td>
</tr>
<tr>
<td>quantity</td>
<td>Double</td>
<td>Quantity of the future inventory.</td>
<td>5.1.0</td>
</tr>
</tbody>
</table>

### ConnectApi.OCIGetInventoryAvailabilityOutputRepresentation

Response to a request for inventory availability data.

Subclass of `ConnectApi.OCIBaseOutputRepresentation`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>locationGroups</td>
<td>List&lt;ConnectApi.OCILocationGroupAvailabilityOutputRepresentation&gt;</td>
<td>A list of inventory availability data for individual location groups.</td>
<td>5.1.0</td>
</tr>
</tbody>
</table>
# Inventory Availability Data

## ConnectApi.OCIInventoryRecordOutputRepresentation

Inventory availability data for a product.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>availableToFulfill</td>
<td>Double</td>
<td>The Available To Fulfill quantity.</td>
<td>51.0</td>
</tr>
<tr>
<td>availableToOrder</td>
<td>Double</td>
<td>The Available To Order quantity.</td>
<td>51.0</td>
</tr>
<tr>
<td>effectiveDate</td>
<td>Datetime</td>
<td>The effective date of the inventory.</td>
<td>51.0</td>
</tr>
<tr>
<td>futures</td>
<td>List&lt;ConnectApi.OCIFutureInventoryOutputRepresentation&gt;</td>
<td>A list of any expected future inventory restocks.</td>
<td>51.0</td>
</tr>
<tr>
<td>onHand</td>
<td>Double</td>
<td>The On Hand quantity.</td>
<td>51.0</td>
</tr>
<tr>
<td>reserved</td>
<td>Double</td>
<td>The Reserved quantity.</td>
<td>51.0</td>
</tr>
<tr>
<td>safetyStockCount</td>
<td>Double</td>
<td>The Safety Stock Count.</td>
<td>51.0</td>
</tr>
<tr>
<td>stockKeepingUnit</td>
<td>String</td>
<td>The SKU of the product.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

## ConnectApi.OCILocationAvailabilityOutputRepresentation

A set of inventory availability data for one inventory location.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>inventoryRecords</td>
<td>List&lt;ConnectApi.OCIInventoryRecordOutputRepresentation&gt;</td>
<td>A list of availability data for individual products at this location.</td>
<td>51.0</td>
</tr>
<tr>
<td>locationIdentifier</td>
<td>String</td>
<td>The identifier of the location.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

## ConnectApi.OCILocationGroupAvailabilityOutputRepresentation

A set of inventory availability data for one inventory location group.

---

**SEE ALSO:**
- `getInventoryAvailability(inventoryAvailabilityInputRepresentation)`
- `findRoutesWithFewestSplitsUsingOCI(findRoutesWithFewestSplitsUsingOCIInput)`
- `ConnectApi.FindRoutesWithFewestSplitsUsingOCIOutputRepresentation`
### ConnectApi.OCIInventoryRecordOutputRepresentation

A list of availability data for individual products. The data combines the quantities for all locations belonging to this location group.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>inventoryRecords</td>
<td>List&lt;ConnectApi.OCIInventoryRecordOutputRepresentation&gt;</td>
<td>A list of availability data for individual products. The data combines the quantities for all locations belonging to this location group.</td>
<td>51.0</td>
</tr>
<tr>
<td>locationGroup</td>
<td>String</td>
<td>The identifier of the location group.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

### ConnectApi.OCIInventoryRecordResponse

Response to a request to publish inventory data.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>inventoryRecords</td>
<td>List&lt;ConnectApi.OCIInventoryRecordOutputRepresentation&gt;</td>
<td>A list of availability data for individual products. The data combines the quantities for all locations belonging to this location group.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

### ConnectApi.OCIInventoryRecordResponseStatus

Detailed status of a publish inventory data job.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>endTimeUTC</td>
<td>String</td>
<td>The UTC time when the job finished. (for example: &quot;2020-07-06T22:54:08.012Z&quot;)</td>
<td>51.0</td>
</tr>
<tr>
<td>recordsProcessedCount</td>
<td>Integer</td>
<td>The number of records processed by the job.</td>
<td>51.0</td>
</tr>
<tr>
<td>recordsReadCount</td>
<td>Integer</td>
<td>The number of records read by the job.</td>
<td>51.0</td>
</tr>
<tr>
<td>recordsSkippedCount</td>
<td>Integer</td>
<td>The number of records skipped by the job.</td>
<td>51.0</td>
</tr>
<tr>
<td>startTimeUTC</td>
<td>String</td>
<td>The UTC time when the job started. (for example: &quot;2020-07-06T22:53:06.788Z&quot;)</td>
<td>51.0</td>
</tr>
<tr>
<td>status</td>
<td>String</td>
<td>The status of the job. (e.g., &quot;PENDING,&quot; &quot;COMPLETED,&quot; etc.).</td>
<td>51.0</td>
</tr>
<tr>
<td>uploadId</td>
<td>String</td>
<td>Identifier of the job.</td>
<td>51.0</td>
</tr>
<tr>
<td>validationErrors</td>
<td>List&lt;String&gt;</td>
<td>List of any validation errors returned by the job.</td>
<td>51.0</td>
</tr>
<tr>
<td>validationStatus</td>
<td>String</td>
<td>The validation status of the job.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

### ConnectApi.OCIInventoryRecordResponseStatus

Response to a request to finalize inventory data.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>endTimeUTC</td>
<td>String</td>
<td>The UTC time when the job finished. (for example: &quot;2020-07-06T22:54:08.012Z&quot;)</td>
<td>51.0</td>
</tr>
<tr>
<td>recordsProcessedCount</td>
<td>Integer</td>
<td>The number of records processed by the job.</td>
<td>51.0</td>
</tr>
<tr>
<td>recordsReadCount</td>
<td>Integer</td>
<td>The number of records read by the job.</td>
<td>51.0</td>
</tr>
<tr>
<td>recordsSkippedCount</td>
<td>Integer</td>
<td>The number of records skipped by the job.</td>
<td>51.0</td>
</tr>
<tr>
<td>startTimeUTC</td>
<td>String</td>
<td>The UTC time when the job started. (for example: &quot;2020-07-06T22:53:06.788Z&quot;)</td>
<td>51.0</td>
</tr>
<tr>
<td>status</td>
<td>String</td>
<td>The status of the job. (e.g., &quot;PENDING,&quot; &quot;COMPLETED,&quot; etc.).</td>
<td>51.0</td>
</tr>
<tr>
<td>uploadId</td>
<td>String</td>
<td>Identifier of the job.</td>
<td>51.0</td>
</tr>
<tr>
<td>validationErrors</td>
<td>List&lt;String&gt;</td>
<td>List of any validation errors returned by the job.</td>
<td>51.0</td>
</tr>
<tr>
<td>validationStatus</td>
<td>String</td>
<td>The validation status of the job.</td>
<td>51.0</td>
</tr>
</tbody>
</table>
### ConnectApi.OCIReleaseReservationOutputRepresentation

Response to a request to release one or more inventory reservations.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>errors</td>
<td>List&lt;ConnectApi.OCIReleaseReservationErrorOutputRepresentation&gt;</td>
<td>Responses for the individual reservations in the release request.</td>
<td>51.0</td>
</tr>
<tr>
<td>success</td>
<td>Boolean</td>
<td>Indicates whether the request was successful.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

### ConnectApi.OCIReleaseReservationSingleOutputRepresentation

Details of a single released reservation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>actionRequestId</td>
<td>String</td>
<td>The UUID that identifies the original release reservation request.</td>
<td>51.0</td>
</tr>
<tr>
<td>externalRefId</td>
<td>String</td>
<td>The external reference ID of the location that released the reservation.</td>
<td>51.0</td>
</tr>
<tr>
<td>locationGroupIdentifier</td>
<td>String</td>
<td>The identifier of the location group that released the reservation.</td>
<td>51.0</td>
</tr>
<tr>
<td>locationIdentifier</td>
<td>String</td>
<td>The identifier of the location that released the reservation.</td>
<td>51.0</td>
</tr>
<tr>
<td>quantity</td>
<td>Double</td>
<td>The released quantity.</td>
<td>51.0</td>
</tr>
<tr>
<td>stockKeepingUnit</td>
<td>String</td>
<td>The SKU of the released product.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

### ConnectApi.OCITransferReservationErrorOutputRepresentation

Response to a request to fulfill one inventory reservation.
### ConnectApi.OCITransferReservationOutputRepresentation

Response to a request to transfer one or more inventory reservations.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>errors</td>
<td>List&lt;ConnectApi.OCITransferReservationErrorOutputRepresentation&gt;</td>
<td>Responses for the individual reservations in the transfer request.</td>
<td>51.0</td>
</tr>
<tr>
<td>success</td>
<td>Boolean</td>
<td>Indicates whether the request was successful.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

### ConnectApi.OCITransferReservationSingleOutputRepresentation

Details of a single transferred reservation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>actionRequestId</td>
<td>String</td>
<td>The UUID that identifies the original transfer reservation request.</td>
<td>51.0</td>
</tr>
<tr>
<td>externalRefId</td>
<td>String</td>
<td>The external reference ID of the location that received the reservation.</td>
<td>51.0</td>
</tr>
<tr>
<td>fromLocationGroupIdentifier</td>
<td>String</td>
<td>The identifier of the location group that sent the reservation.</td>
<td>51.0</td>
</tr>
<tr>
<td>fromLocationIdentifier</td>
<td>String</td>
<td>The identifier of the location that sent the reservation.</td>
<td>51.0</td>
</tr>
<tr>
<td>ignoreAvailabilityCheck</td>
<td>Boolean</td>
<td>Whether this call ignored availability data at the location that received the reservation.</td>
<td>51.0</td>
</tr>
<tr>
<td>quantity</td>
<td>Double</td>
<td>The quantity of transferred inventory.</td>
<td>51.0</td>
</tr>
<tr>
<td>stockKeepingUnit</td>
<td>String</td>
<td>The SKU of the transferred product.</td>
<td>51.0</td>
</tr>
<tr>
<td>toLocationGroupIdentifier</td>
<td>String</td>
<td>The identifier of the location group that received the reservation.</td>
<td>51.0</td>
</tr>
<tr>
<td>toLocationIdentifier</td>
<td>String</td>
<td>The identifier of the location that received the reservation.</td>
<td>51.0</td>
</tr>
</tbody>
</table>
### ConnectApi.OCIUploadInventoryAvailabilityOutputRepresentation

Response to an upload inventory availability job.
Subclass of ConnectApi.OCIBaseOutputRepresentation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>uploadId</td>
<td>String</td>
<td>Identifier of the upload job. Use this value to retrieve the status of the job.</td>
<td>5.1.0</td>
</tr>
</tbody>
</table>

### ConnectApi.OCIUploadInventoryAvailabilityStatusOutputRepresentation

Detailed status of an upload inventory availability job.
Subclass of ConnectApi.OCIBaseOutputRepresentation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>endTimeUTC</td>
<td>String</td>
<td>The UTC time when the job finished. (for example: &quot;2020-07-06T22:54:08.012Z&quot;)</td>
<td>5.1.0</td>
</tr>
<tr>
<td>recordsProcessedCount</td>
<td>Integer</td>
<td>The number of records processed by the job.</td>
<td>5.1.0</td>
</tr>
<tr>
<td>recordsReadCount</td>
<td>Integer</td>
<td>The number of records read by the job.</td>
<td>5.1.0</td>
</tr>
<tr>
<td>recordsSkippedCount</td>
<td>Integer</td>
<td>The number of records skipped by the job.</td>
<td>5.1.0</td>
</tr>
<tr>
<td>startTimeUTC</td>
<td>String</td>
<td>The UTC time when the job started. (for example: &quot;2020-07-06T22:53:06.788Z&quot;)</td>
<td>5.1.0</td>
</tr>
<tr>
<td>status</td>
<td>String</td>
<td>The status of the job. (e.g., &quot;PENDING,&quot; &quot;COMPLETED,&quot; etc.).</td>
<td>5.1.0</td>
</tr>
<tr>
<td>uploadId</td>
<td>String</td>
<td>Identifier of the job.</td>
<td>5.1.0</td>
</tr>
<tr>
<td>validationErrors</td>
<td>List&lt;String&gt;</td>
<td>List of any validation errors returned by the job.</td>
<td>5.1.0</td>
</tr>
<tr>
<td>validationStatus</td>
<td>String</td>
<td>The validation status of the job.</td>
<td>5.1.0</td>
</tr>
</tbody>
</table>

### ConnectApi.OrchestrationInstance

Orchestration instance.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>flowDefinition</td>
<td>String</td>
<td>Developer name of the flow definition.</td>
<td>5.4.0</td>
</tr>
<tr>
<td>DeveloperName</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>flowDefinitionId</td>
<td>String</td>
<td>ID of the flow definition.</td>
<td>5.4.0</td>
</tr>
<tr>
<td>flowDefinitionName</td>
<td>String</td>
<td>Name of the flow definition.</td>
<td>5.4.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the orchestration instance.</td>
<td>5.4.0</td>
</tr>
<tr>
<td>interviewId</td>
<td>String</td>
<td>ID of the interview to resume.</td>
<td>5.4.0</td>
</tr>
</tbody>
</table>
### ConnectApi.OrchestrationStageInstance
Orchestration stage instance.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the orchestration stage instance.</td>
<td>54.0</td>
</tr>
<tr>
<td>label</td>
<td>String</td>
<td>Orchestration stage instance label.</td>
<td>54.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Orchestration stage instance name.</td>
<td>54.0</td>
</tr>
<tr>
<td>status</td>
<td>ConnectApi.OrchestrationStatus</td>
<td>Status of the orchestration instance. Values are:</td>
<td>54.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Completed</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- InProgress</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- NotStarted</td>
<td></td>
</tr>
<tr>
<td>stepInstances</td>
<td>List&lt;ConnectApi.OrchestrationStepInstance&gt;</td>
<td>Orchestration stage instance steps.</td>
<td>54.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.OrchestrationInstance

### ConnectApi.OrchestrationInstanceCollection
Collection of orchestration instances.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>instances</td>
<td>List&lt;ConnectApi.OrchestrationInstance&gt;</td>
<td>Collection of orchestration instances.</td>
<td>54.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.OrchestrationInstanceCollection
**ConnectApi.OrchestrationStepInstance**

Orchestration step instance.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the orchestration step instance.</td>
<td>54.0</td>
</tr>
<tr>
<td>label</td>
<td>String</td>
<td>Orchestration step instance label.</td>
<td>54.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Orchestration step instance name.</td>
<td>54.0</td>
</tr>
<tr>
<td>status</td>
<td>ConnectApi.OrchestrationStatus</td>
<td>Status of the orchestration instance. Values are:</td>
<td>54.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Completed</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• InProgress</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NotStarted</td>
<td></td>
</tr>
<tr>
<td>stepType</td>
<td>ConnectApi.OrchestrationStepType</td>
<td>Type of orchestration step. Values are:</td>
<td>54.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AsynchronousBackgroundStep</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• BackgroundStep</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• InteractiveStep</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ManagedContentRoleInteractiveStep</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ManagedContentVariantAutoPublishBackgroundStep</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ManagedContentVariantAutoUnpublishBackgroundStep</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ManagedContentVariantSetLockBackgroundStep</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ManagedContentVariantSetReadyBackgroundStep</td>
<td></td>
</tr>
<tr>
<td>workItems</td>
<td>List&lt;ConnectApi.OrchestrationWorkItem&gt;</td>
<td>Orchestration step instance work items.</td>
<td>54.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

*ConnectApi.OrchestrationStageInstance*

**ConnectApi.OrchestrationWorkItem**

Orchestration work item.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>assigneeId</td>
<td>String</td>
<td>ID of the assignee for the orchestration work item.</td>
<td>54.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the orchestration work item.</td>
<td>54.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the orchestration work item.</td>
<td>54.0</td>
</tr>
<tr>
<td>label</td>
<td>String</td>
<td>Label key for the orchestration work item.</td>
<td>54.0</td>
</tr>
</tbody>
</table>
### ConnectApi.OrchestrationStepInstance

Order delivery group summary.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>fields</td>
<td>Map&lt;String, ConnectApi.RecordField&gt;</td>
<td>Map of fields from order delivery group summary and other related objects that were queried.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.OrderDeliveryGroupSummaryCollection

### ConnectApi.OrderDeliveryGroupSummaryCollection

Collection of order delivery group summaries.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageToken</td>
<td>String</td>
<td>Token identifying the current page of order delivery group summaries.</td>
<td>51.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>URL to the current page of order delivery group summaries.</td>
<td>51.0</td>
</tr>
<tr>
<td>nextPageToken</td>
<td>String</td>
<td>Token identifying the next page of order delivery group summaries.</td>
<td>51.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>URL to the next page of order delivery group summaries.</td>
<td>51.0</td>
</tr>
</tbody>
</table>
### ConnectApi.OrderDeliveryGroupSummaryLookupOutput

Order delivery group summary lookup output.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Three-letter ISO 4217 currency code associated with the order delivery group summary record.</td>
<td>58.0</td>
</tr>
<tr>
<td>deliveryMethod</td>
<td>ConnectApi.OrderDeliveryMethodLookupOutput</td>
<td>Delivery method associated with order the delivery group summary.</td>
<td>58.0</td>
</tr>
<tr>
<td>fields</td>
<td>Map&lt;String, ConnectApi.RecordField&gt;</td>
<td>Map of requested order delivery group summary fields.</td>
<td>58.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the order delivery group summary.</td>
<td>58.0</td>
</tr>
<tr>
<td>lineItems</td>
<td>ConnectApi.OrderSummaryLookupOutput</td>
<td>Line items associated with the order delivery group summary.</td>
<td>58.0</td>
</tr>
</tbody>
</table>

### ConnectApi.OrderDeliveryMethodLookupOutput

Order delivery method lookup output.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>fields</td>
<td>Map&lt;String, ConnectApi.RecordField&gt;</td>
<td>Map of requested order delivery method fields.</td>
<td>58.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the order delivery method.</td>
<td>58.0</td>
</tr>
</tbody>
</table>

### ConnectApi.OrderItemSummary

Order item summary.
## ConnectApi.OrderItemSummaryAdjustmentAggregates

Adjustment aggregates associated with an order item summary.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>available</td>
<td>Boolean</td>
<td>Indicates whether adjustment aggregates are available (true) or not (false).</td>
<td>55.0</td>
</tr>
<tr>
<td>status</td>
<td>ConnectApi.OrderSummaryAdjustmentAggregatesStatus</td>
<td>Order summary adjustment aggregate job status.</td>
<td>55.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Failed—The adjustment aggregate data job for the order summary failed.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• InProgress—The adjustment aggregate data job for the order summary is in progress.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NotInitiated—The adjustment aggregate data job for the order summary is not initiated.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Submitted—The adjustment aggregate data job for the order summary is submitted.</td>
<td></td>
</tr>
<tr>
<td>totalLinePromotionAmount</td>
<td>String</td>
<td>Total of all line item promotions applied to this specific product.</td>
<td>55.0</td>
</tr>
<tr>
<td>totalPromotionDistAmount</td>
<td>String</td>
<td>Total of all order level promotions applied to this specific product.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

### SEE ALSO:

- ConnectApi.OrderItemSummaryCollection
ConnectApi.OrderItemSummaryAdjustmentCollection

Collection of adjustments for order item summaries.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>orderItemSummaries</td>
<td>Map&lt;String, ConnectApi.OrderItemSummaryAdjustmentList&gt;</td>
<td>Order item summaries and their associated adjustments.</td>
<td>53.0</td>
</tr>
</tbody>
</table>

ConnectApi.OrderItemSummaryAdjustmentList

Representation for list of adjustments for an Order Item Summary.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>adjustments</td>
<td>List&lt;ConnectApi.OrderSummaryAdjustment&gt;</td>
<td>Adjustments associated with an order item summary.</td>
<td>53.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

ConnectApi.OrderItemSummaryAdjustmentCollection

ConnectApi.OrderItemSummaryCollection

Collection of order item summaries.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageToken</td>
<td>String</td>
<td>Token identifying the current page of items.</td>
<td>51.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>URL to the current page of items.</td>
<td>51.0</td>
</tr>
<tr>
<td>items</td>
<td>List&lt;ConnectApi.OrderItemSummary&gt;</td>
<td>Collection of order item summaries.</td>
<td>51.0</td>
</tr>
<tr>
<td>nextPageToken</td>
<td>String</td>
<td>Token identifying the next page of items.</td>
<td>51.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>URL to the next page of items.</td>
<td>51.0</td>
</tr>
<tr>
<td>previousPageToken</td>
<td>String</td>
<td>Token identifying the previous page of items.</td>
<td>51.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>URL to the previous page of items.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

ConnectApi.OrderItemSummaryLookupOutput

Order item summary lookup output.
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>adjustmentAggregates</td>
<td>ConnectApi.OrderItemSummaryAdjustmentAggregates</td>
<td>Adjustment aggregates for the order item summary.</td>
<td>58.0</td>
</tr>
<tr>
<td>adjustments</td>
<td>List&lt;ConnectApi.OrderSummaryAdjustment&gt;</td>
<td>Adjustments associated with the order item summary.</td>
<td>58.0</td>
</tr>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Three-letter ISO 4217 currency code associated with the order item summary record.</td>
<td>58.0</td>
</tr>
<tr>
<td>fields</td>
<td>Map&lt;String, ConnectApi.RecordField&gt;</td>
<td>Map of requested order item summary fields.</td>
<td>58.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the order item summary.</td>
<td>58.0</td>
</tr>
<tr>
<td>product</td>
<td>ConnectApi.OrderItemSummaryProduct</td>
<td>Details of the product associated with order item summary.</td>
<td>58.0</td>
</tr>
</tbody>
</table>

**ConnectApi.OrderItemSummaryOutputRepresentation**

Details of an OrderItemSummary from a failed FulfillmentOrder in a create multiple fulfillment orders request.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>errors</td>
<td>List&lt;ConnectApi.ErrorResponse&gt;</td>
<td>List of errors specific to the OrderItemSummary, if any.</td>
<td>50.0</td>
</tr>
<tr>
<td>orderItemSummaryId</td>
<td>String</td>
<td>ID of the OrderItemSummary.</td>
<td>50.0</td>
</tr>
<tr>
<td>quantity</td>
<td>Double</td>
<td>Quantity of the OrderItemSummary.</td>
<td>50.0</td>
</tr>
</tbody>
</table>

**ConnectApi.OrderItemSummaryProduct**

Product item mapped to the order item summary.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>canViewProduct</td>
<td>Boolean</td>
<td>Specifies whether the context user can view the product (true) or not (false).</td>
<td>51.0</td>
</tr>
<tr>
<td>errorCode</td>
<td>String</td>
<td>Error code for the product with errors.</td>
<td>51.0</td>
</tr>
<tr>
<td>errorMessage</td>
<td>String</td>
<td>Error message for the product with errors.</td>
<td>51.0</td>
</tr>
<tr>
<td>fields</td>
<td>Map&lt;String, ConnectApi.RecordField&gt;</td>
<td>Map of the product fields queried.</td>
<td>51.0</td>
</tr>
<tr>
<td>media</td>
<td>ConnectApi.ProductMedia</td>
<td>Associated product media.</td>
<td>51.0</td>
</tr>
<tr>
<td>productAttributes</td>
<td>ConnectApi.ProductAttributeSetSummary</td>
<td>Summary of the product attributes.</td>
<td>51.0</td>
</tr>
</tbody>
</table>
### ConnectApi.Product

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>productId</td>
<td>String</td>
<td>ID of the product.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

- [ConnectApi.OrderItemSummary](#)

---

### ConnectApi.OrderShipment

Order shipment.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>expectedDeliveryDate</td>
<td>DateTime</td>
<td>Expected delivery date for the shipment.</td>
<td>52.0</td>
</tr>
<tr>
<td>fields</td>
<td>Map&lt;String, ConnectApi.RecordField&gt;</td>
<td>Map of requested fields.</td>
<td>52.0</td>
</tr>
<tr>
<td>orderSummaryId</td>
<td>String</td>
<td>ID of the order summary.</td>
<td>52.0</td>
</tr>
<tr>
<td>shipmentId</td>
<td>String</td>
<td>ID of the shipment.</td>
<td>52.0</td>
</tr>
<tr>
<td>shipmentNumber</td>
<td>String</td>
<td>Number of the shipment.</td>
<td>52.0</td>
</tr>
<tr>
<td>status</td>
<td>String</td>
<td>Status of the shipment.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

- [ConnectApi.OrderShipmentCollection](#)

---

### ConnectApi.OrderShipmentCollection

Collection of order shipments.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>count</td>
<td>Integer</td>
<td>Total number of records returned in the collection.</td>
<td>52.0</td>
</tr>
<tr>
<td>currentPageToken</td>
<td>String</td>
<td>Token identifying the current page of order shipments.</td>
<td>52.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>URL to the current page of order shipments.</td>
<td>52.0</td>
</tr>
<tr>
<td>nextPageToken</td>
<td>String</td>
<td>Token identifying the next page of order shipments.</td>
<td>52.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>URL to the next page of order shipments.</td>
<td>52.0</td>
</tr>
<tr>
<td>previousPageToken</td>
<td>String</td>
<td>Token identifying the previous page of order shipments.</td>
<td>52.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>URL to the previous page of order shipments.</td>
<td>52.0</td>
</tr>
</tbody>
</table>
### ConnectApi.OrderShipment

Shipment item.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>shipments</td>
<td>List&lt;ConnectApi.OrderShipment&gt;</td>
<td>Collection of order shipments.</td>
<td>52.0</td>
</tr>
<tr>
<td>sortOrder</td>
<td>ConnectApi.OrderShipmentSort</td>
<td>Sort order for order shipments. Values are:</td>
<td>52.0</td>
</tr>
<tr>
<td></td>
<td>• ExpectedDeliveryDateAsc—Sorts by</td>
<td>the oldest expected delivery date.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>the most recent expected delivery</td>
<td>date.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• ShipmentNumberAsc—Sorts by shipment</td>
<td>number in ascending order (0–9).</td>
<td></td>
</tr>
<tr>
<td></td>
<td>number in descending order (9–0).</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### ConnectApi.OrderShipmentItem

Collection of order shipment items.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the order item summary.</td>
<td>52.0</td>
</tr>
<tr>
<td>productId</td>
<td>String</td>
<td>ID of the product.</td>
<td>52.0</td>
</tr>
<tr>
<td>quantity</td>
<td>Double</td>
<td>Quantity of the product.</td>
<td>52.0</td>
</tr>
<tr>
<td>shipmentId</td>
<td>String</td>
<td>ID of the shipment.</td>
<td>52.0</td>
</tr>
<tr>
<td>shipmentItemId</td>
<td>String</td>
<td>ID of the shipment item.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

### ConnectApi.OrderShipmentItemCollection

Collection of order shipment items.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>count</td>
<td>Integer</td>
<td>Total number of records returned in a page.</td>
<td>52.0</td>
</tr>
<tr>
<td>currentPageToken</td>
<td>String</td>
<td>Token identifying the current page of order shipment items.</td>
<td>52.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>URL to the current page of order shipment items.</td>
<td>52.0</td>
</tr>
</tbody>
</table>
### ConnectApi.OrderSummaryLookupOutput

Order summary lookup output.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>adjustmentAggregates</td>
<td>ConnectApi.OrderSummaryAdjustmentAggregates</td>
<td>Adjustment aggregates associated with the order summary.</td>
<td>58.0</td>
</tr>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Three-letter ISO 4217 currency code associated with the order summary.</td>
<td>58.0</td>
</tr>
<tr>
<td>deliveryGroups</td>
<td>ConnectApi.OrderSummaryDeliveryGroupLookupOutput</td>
<td>Delivery groups associated with the order summary.</td>
<td>58.0</td>
</tr>
<tr>
<td>fields</td>
<td>Map&lt;String, ConnectApi.RecordField&gt;</td>
<td>Map of requested order summary fields.</td>
<td>58.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the order summary.</td>
<td>58.0</td>
</tr>
<tr>
<td>orderNumber</td>
<td>String</td>
<td>Reference number of the order summary.</td>
<td>58.0</td>
</tr>
<tr>
<td>status</td>
<td>String</td>
<td>Status associated with the order summary.</td>
<td>58.0</td>
</tr>
</tbody>
</table>

### ConnectApi.OrderSummaryOutputRepresentation

ID of the created Order Summary.

Subclass of ConnectApi.BaseOutputRepresentation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>orderSummaryId</td>
<td>String</td>
<td>ID of the Order Summary.</td>
<td>48.0</td>
</tr>
</tbody>
</table>
### ConnectApi.OrderSummaryProductLookupOutput

Order summary product lookup output.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>canViewProduct</td>
<td>Boolean</td>
<td>Specifies whether the context user can view the product (true) or not (false).</td>
<td>58.0</td>
</tr>
<tr>
<td>errorCode</td>
<td>String</td>
<td>Error code captured during product load.</td>
<td>58.0</td>
</tr>
<tr>
<td>errorMessage</td>
<td>String</td>
<td>Error message captured during product load.</td>
<td>58.0</td>
</tr>
<tr>
<td>fields</td>
<td>Map&lt;String, ConnectApi.RecordField&gt;</td>
<td>Map of requested product fields.</td>
<td>58.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>Id of the product</td>
<td>58.0</td>
</tr>
<tr>
<td>media</td>
<td>ConnectApi.ProductMedia</td>
<td>Associated product media.</td>
<td>58.0</td>
</tr>
<tr>
<td>variationAttributes</td>
<td>Map&lt;String, ConnectApi.OrderSummaryProductAttribute&gt;</td>
<td>Variation attributes (color, size, and so on) associated with the product.</td>
<td>58.0</td>
</tr>
</tbody>
</table>

### ConnectApi.OrderSummaryRepresentation

Order summary.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>adjustmentAggregates</td>
<td>ConnectApi.OrderSummaryAdjustmentAggregates</td>
<td>Adjustment aggregates associated with the order summary.</td>
<td>55.0</td>
</tr>
<tr>
<td>createdDate</td>
<td>Datetime</td>
<td>Created date of the order summary.</td>
<td>51.0</td>
</tr>
<tr>
<td>fields</td>
<td>Map&lt;String, ConnectApi.RecordField&gt;</td>
<td>Map of requested order summary fields.</td>
<td>51.0</td>
</tr>
<tr>
<td>orderNumber</td>
<td>String</td>
<td>Order number of the order summary.</td>
<td>51.0</td>
</tr>
<tr>
<td>orderSummaryId</td>
<td>String</td>
<td>ID of the order summary.</td>
<td>51.0</td>
</tr>
<tr>
<td>orderedDate</td>
<td>Datetime</td>
<td>Ordered date of the order summary.</td>
<td>51.0</td>
</tr>
<tr>
<td>ownerId</td>
<td>String</td>
<td>ID of the owner of the order summary.</td>
<td>51.0</td>
</tr>
<tr>
<td>status</td>
<td>String</td>
<td>Status of the order summary.</td>
<td>51.0</td>
</tr>
<tr>
<td>totalAmount</td>
<td>String</td>
<td>Total amount of the order summary.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

ConnectApi.OrderSummaryCollectionRepresentation
## ConnectApi.OrderSummaryAdjustment

Adjustment associated with an order summary.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>amount</td>
<td>String</td>
<td>Amount associated with the adjustment.</td>
<td>53.0</td>
</tr>
<tr>
<td>basisReferenceDisplayName</td>
<td>String</td>
<td>Display name for secondary cause of the adjustment (for example, Null or the CouponCode that’s associated with a Coupon)</td>
<td>54.0</td>
</tr>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Three-letter ISO 4217 currency code associated with the adjustment.</td>
<td>53.0</td>
</tr>
<tr>
<td>displayName</td>
<td>String</td>
<td>Display name for the primary cause of the adjustment (for example, Display name of the Promotion)</td>
<td>53.0</td>
</tr>
<tr>
<td>targetType</td>
<td>ConnectApi.OrderSummaryAdjustmentTargetType</td>
<td>Type of price adjustment in promotions. Values are:</td>
<td>56.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• SplitLine—Price adjustment on an order item.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Header—Price adjustment on the entire order.</td>
<td></td>
</tr>
<tr>
<td>type</td>
<td>String</td>
<td>Type of adjustment (for example, Promotion, Other).</td>
<td>53.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

- ConnectApi.OrderSummaryAdjustmentCollection
- ConnectApi.OrderItemSummaryAdjustmentList

## ConnectApi.OrderSummaryAdjustmentAggregates

Adjustment aggregates associated with an order summary.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>available</td>
<td>Boolean</td>
<td>Indicates if adjustment aggregate values are available (true) or not (false).</td>
<td>55.0</td>
</tr>
<tr>
<td>status</td>
<td>ConnectApi.OrderSummaryAdjustmentAggregatesStatus</td>
<td>Order summary adjustment aggregate job status. Values are:</td>
<td>55.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Failed—The adjustment aggregate data job for the order summary failed.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• InProgress—The adjustment aggregate data job for the order summary is in progress.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NotInitiated—The adjustment aggregate data job for the order summary is not initiated.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Submitted—The adjustment aggregate data job for the order summary is submitted.</td>
<td></td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>----------------------------</td>
<td>-------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>totalDeliveryPromotionDistAmount</td>
<td>String</td>
<td>Total distributed delivery promotion amounts associated with an order summary.</td>
<td>55.0</td>
</tr>
<tr>
<td>totalDeliveryPromotionLineAmount</td>
<td>String</td>
<td>Total delivery promotion line amounts associated with an order summary.</td>
<td>55.0</td>
</tr>
<tr>
<td>totalDeliveryPromotionTotalAmount</td>
<td>String</td>
<td>Total delivery promotion amount associated with an order summary.</td>
<td>55.0</td>
</tr>
<tr>
<td>totalProductPromotionDistAmount</td>
<td>String</td>
<td>Total distributed product promotion amounts associated with an order summary.</td>
<td>55.0</td>
</tr>
<tr>
<td>totalProductPromotionLineAmount</td>
<td>String</td>
<td>Total product promotion line amount associated with an order summary.</td>
<td>55.0</td>
</tr>
<tr>
<td>totalProductPromotionTotalAmount</td>
<td>String</td>
<td>Total product promotion amount associated with an order summary.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

**ConnectApi.OrderSummaryAdjustmentAggregatesAsyncOutput**

Async adjustment aggregates output.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>statusURL</td>
<td>String</td>
<td>Status URL.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

**ConnectApi.OrderSummaryAdjustmentCollection**

Collection of adjustments for an order summary.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>adjustments</td>
<td>List&lt;ConnectApi.OrderSummaryAdjustment&gt;</td>
<td>Collection of adjustments for an order summary.</td>
<td>53.0</td>
</tr>
</tbody>
</table>

**ConnectApi.OrderSummaryCollectionRepresentation**

Collection of order summaries.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>count</td>
<td>Integer</td>
<td>Total count of order summaries returned on the current page.</td>
<td>51.0</td>
</tr>
<tr>
<td>currentPageToken</td>
<td>String</td>
<td>Token identifying the current page.</td>
<td>51.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>-----------------</td>
<td>------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>51.0</td>
</tr>
<tr>
<td>nextPageToken</td>
<td>String</td>
<td>Token identifying the next page, or null if there isn’t a next page.</td>
<td>51.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or null if there isn’t a next page.</td>
<td>51.0</td>
</tr>
<tr>
<td>orderSummaries</td>
<td>List&lt;ConnectApi.OrderSummary Representation&gt;</td>
<td>Collection of order summaries.</td>
<td>51.0</td>
</tr>
<tr>
<td>previousPageToken</td>
<td>String</td>
<td>Token identifying the previous page, or null if there isn’t a previous page.</td>
<td>51.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the previous page, or null if there isn’t a previous page.</td>
<td>51.0</td>
</tr>
</tbody>
</table>
| sortOrder       | ConnectApi.OrderSummary SortOrder | Sort order for order summaries. Values are:  
|                 |                  | - CreatedDateAsc—Sorts by the oldest created date.                        | 51.0              |
|                 |                  | - CreatedDateDesc—Sorts by the most recent created date.                   |                   |
|                 |                  | - OrderedDateAsc—Sorts by the oldest ordered date.                         |                   |
|                 |                  | - OrderedDateDesc—Sorts by the most recent ordered date.                   |                   |

**ConnectApi.OrderSummaryProductAttribute**

Order summary product attribute representation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>label</td>
<td>String</td>
<td>Label or display name of the attribute.</td>
<td>58.0</td>
</tr>
<tr>
<td>sequence</td>
<td>Integer</td>
<td>Sequence of the attribute set with regard to the product.</td>
<td>58.0</td>
</tr>
<tr>
<td>value</td>
<td>String</td>
<td>Display value of the attribute.</td>
<td>58.0</td>
</tr>
</tbody>
</table>

**ConnectApi.OrderToCartFailedProduct**

Product that could not be added to the cart from an order, with error information.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>errorCode</td>
<td>String</td>
<td>Error code.</td>
<td>57.0</td>
</tr>
</tbody>
</table>
### ConnectApi.OrderToCartResult

Result of action adding an order to a cart.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>cartId</td>
<td>String</td>
<td>ID of the cart.</td>
<td>57.0</td>
</tr>
<tr>
<td>totalFailedProductCnt</td>
<td>Integer</td>
<td>Number of products that could not be successfully added to the cart from the order.</td>
<td>57.0</td>
</tr>
<tr>
<td>totalSucceededProductCnt</td>
<td>Integer</td>
<td>Number of products successfully added to the cart from the order.</td>
<td>57.0</td>
</tr>
<tr>
<td>unaddedProducts</td>
<td>ConnectApi.OrderToCartFailedProduct</td>
<td>List of products not successfully added to the cart.</td>
<td>57.0</td>
</tr>
</tbody>
</table>

### ConnectApi.OrganizationSettings

Org settings.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>accessTimeout</td>
<td>Integer</td>
<td>Amount of time after which the system prompts users who have been inactive to log out or continue working.</td>
<td>28.0</td>
</tr>
<tr>
<td>features</td>
<td>ConnectApi.Features</td>
<td>Information about features available in the org.</td>
<td>28.0</td>
</tr>
<tr>
<td>maintenanceInfo</td>
<td>List&lt;ConnectApi.MaintenanceInfo&gt;</td>
<td>Information about a list of upcoming scheduled maintenances for the org.</td>
<td>34.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Org name.</td>
<td>28.0</td>
</tr>
<tr>
<td>orgId</td>
<td>String</td>
<td>18-character ID for the org.</td>
<td>28.0</td>
</tr>
<tr>
<td>userSettings</td>
<td>ConnectApi.UserSettings</td>
<td>Information about the org permissions for the user.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

### ConnectApi-OriginCapability

If a feed element has this capability, it was created by a feed action.

Subclass of ConnectApi.FeedElementCapability.
### ConnectApi.outputClasses

**Actor**

The user who executed the feed action.

**OriginRecord**

A reference to the feed element containing the feed action.

---

**ConnectApi.OutOfOffice**

User's out-of-office message.

**Message**

Out-of-office message for the user.

---

**Important:** Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

**ConnectApi.PardotBusinessUnitContextItem**

Pardot business unit context item.

**Id**

ID of the PardotTenant record.

**IsCurrent**

Specifies whether the business unit is selected as the context user’s current business unit context in the business unit switcher of the Pardot Lightning app (true) or not (false).

**Name**

Name of the Pardot business unit as it is specified in the MasterLabel of the PardotTenant record.

---

**ConnectApi.PardotBusinessUnitContextOutput**

Pardot business unit context.
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>businessUnits</td>
<td>List&lt;ConnectApi.PardotBusinessUnitContextItem&gt;</td>
<td>List of the Pardot business unit context items that the context user has access to.</td>
<td>55.0</td>
</tr>
<tr>
<td>isSuccess</td>
<td>Boolean</td>
<td>Indicates whether the requested resource was successfully provided.</td>
<td>55.0</td>
</tr>
<tr>
<td>totalBusinessUnits</td>
<td>Integer</td>
<td>Indicates the total number of Pardot business units that the context user has access to.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

**ConnectApi.PaymentAuthorizationResponse**

Payment authorization output representation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>accountId</td>
<td>String</td>
<td>Salesforce account for the payment authorization.</td>
<td>51.0</td>
</tr>
<tr>
<td>amount</td>
<td>Double</td>
<td>Amount that the gateway authorized for the payment transaction.</td>
<td>51.0</td>
</tr>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Three-letter ISO 4217 currency code associated with the payment group record.</td>
<td>51.0</td>
</tr>
<tr>
<td>effectiveDate</td>
<td>Datetime</td>
<td>Date that the authorization becomes effective.</td>
<td>51.0</td>
</tr>
<tr>
<td>expirationDate</td>
<td>Datetime</td>
<td>Date that the authorization expires.</td>
<td>51.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the payment authorization record.</td>
<td>51.0</td>
</tr>
<tr>
<td>paymentAuthorizationNumber</td>
<td>String</td>
<td>System-defined number for the payment authorization record.</td>
<td>51.0</td>
</tr>
<tr>
<td>requestDate</td>
<td>Datetime</td>
<td>Date that the authorization occurred.</td>
<td>51.0</td>
</tr>
<tr>
<td>status</td>
<td>String</td>
<td>Status of the payment authorization as returned by the gateway.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

**ConnectApi.PaymentGroupResponse**

Payment group.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Three-letter ISO 4217 currency code associated with the payment group record.</td>
<td>50.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the payment group record.</td>
<td>50.0</td>
</tr>
<tr>
<td>sourceObjectId</td>
<td>String</td>
<td>Source object ID of the payment group record. Supports only OrderId.</td>
<td>50.0</td>
</tr>
</tbody>
</table>
## ConnectApi.PaymentMethodDetails
Details about the payment method.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>alternativePaymentMethod</td>
<td>ConnectApi.AlternativePaymentMethodOutput</td>
<td>Alternative Payment Method details.</td>
<td>56.0</td>
</tr>
<tr>
<td>cardPaymentMethod</td>
<td>ConnectApi.CardPaymentMethodOutput</td>
<td>Card Payment Method details.</td>
<td>56.0</td>
</tr>
</tbody>
</table>

## ConnectApi.PaymentMethodResponse
Payment method information response.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>accountId</td>
<td>String</td>
<td>Salesforce Payments account to which this payment method is linked.</td>
<td>51.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the payment method.</td>
<td>51.0</td>
</tr>
<tr>
<td>paymentMethodDetails</td>
<td>ConnectApi.PaymentMethodDetails</td>
<td>Details about the payment method.</td>
<td></td>
</tr>
<tr>
<td>status</td>
<td>String</td>
<td>Status of the payment method.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

## ConnectApi.PaymentMethodTokenizationGatewayResponse
Payment method tokenization gateway response representation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>gatewayToken</td>
<td>String</td>
<td>The payment method token sent from the gateway.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

## ConnectApi.PaymentMethodTokenizationResponse
Payment method tokenization output representation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>error</td>
<td>ConnectApi.ErrorResponse</td>
<td>Error representation for the payment method tokenization process. Sent only if the tokenization process encounters an error in the gateway.</td>
<td>52.0</td>
</tr>
<tr>
<td>gatewayResponse</td>
<td>ConnectApi.PaymentMethodTokenizationGatewayResponse</td>
<td>Response containing the tokenized payment method value from the payment gateway.</td>
<td>52.0</td>
</tr>
</tbody>
</table>
### Available Version

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>paymentGatewayLogs</td>
<td><code>List&lt;ConnectApi.GatewayLogResponse&gt;</code></td>
<td>Logs showing more details about the tokenization process that occurred in the gateway.</td>
<td>52.0</td>
</tr>
<tr>
<td>paymentMethod</td>
<td><code>ConnectApi.PaymentMethodResponse</code></td>
<td>Object representation of the payment method object that was tokenized.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

### ConnectApi.PaymentResponse

Payment output.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>accountId</td>
<td><code>String</code></td>
<td>ID of the account related the payment record.</td>
<td>50.0</td>
</tr>
<tr>
<td>amount</td>
<td><code>Double</code></td>
<td>Total amount of the payment transaction performed in the payment request.</td>
<td>50.0</td>
</tr>
<tr>
<td>currencyIsoCode</td>
<td><code>String</code></td>
<td>Three-letter ISO 4217 currency code associated with the payment output.</td>
<td>50.0</td>
</tr>
<tr>
<td>effectiveDate</td>
<td><code>Datetime</code></td>
<td>Date that the payment becomes effective.</td>
<td>50.0</td>
</tr>
<tr>
<td>id</td>
<td><code>String</code></td>
<td>ID of the payment record.</td>
<td>50.0</td>
</tr>
<tr>
<td>paymentNumber</td>
<td><code>String</code></td>
<td>Number of the payment record created as a result of the request processing.</td>
<td>50.0</td>
</tr>
<tr>
<td>requestDate</td>
<td><code>Datetime</code></td>
<td>Date when the payment transaction occurred.</td>
<td>50.0</td>
</tr>
<tr>
<td>status</td>
<td><code>String</code></td>
<td>Status of the new payment record. Can be DRAFT, PROCESSED or CANCELLED.</td>
<td>50.0</td>
</tr>
</tbody>
</table>

### ConnectApi.PercentRecordField

Record field containing a percentage value. Subclass of ConnectApi.LabeledRecordField.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>value</td>
<td><code>Double</code></td>
<td>Value of the percentage.</td>
<td>29.0</td>
</tr>
</tbody>
</table>

### ConnectApi.PhoneNumber

Phone number.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>label</td>
<td><code>String</code></td>
<td>A localized string indicating the phone type.</td>
<td>30.0</td>
</tr>
<tr>
<td>phoneNumber</td>
<td><code>String</code></td>
<td>Phone number.</td>
<td>28.0</td>
</tr>
</tbody>
</table>
### phoneType

**Type:** String

Phone type. Values are:

- Fax
- Mobile
- Work

These values are not localized.

**Available Version:** 30.0

---

**Note:** This property is not available after version 29.0. Use the `phoneType` property instead.

Values are:
- Fax
- Mobile
- Work

These values are not localized.

**Available Version:** 28.0–29.0

---

### ConnectApi.Photo

Profile photo.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>fullEmailPhotoUrl</td>
<td>String</td>
<td>A temporary URL to the large profile picture. The URL expires after 30 days and is available to unauthenticated users.</td>
<td>28.0</td>
</tr>
<tr>
<td>largePhotoUrl</td>
<td>String</td>
<td>URL to the large profile picture. The default width is 200 pixels, and the height is scaled so the original image proportions are maintained. If a user hasn’t uploaded a photo, this URL points to a default photo. If the user hasn’t uploaded a photo and the request header included <code>X-Connect-Theme: Salesforce1</code>, this URL points to a default photo based on a theme that the admin selected for the org.</td>
<td>28.0</td>
</tr>
<tr>
<td>mediumPhotoUrl</td>
<td>String</td>
<td>URL to the medium profile picture. The default width is 160 pixels, and the height is scaled so the original image proportions are maintained. If a user hasn’t uploaded a photo, this URL points to a default photo.</td>
<td>37.0</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>photoVersionId</td>
<td>String</td>
<td>18-character ID to that version of the photo</td>
<td>28.0</td>
</tr>
<tr>
<td>smallPhotoUrl</td>
<td>String</td>
<td>URL to the small profile picture. The default size is 64x64 pixels. If a user hasn’t uploaded a photo, this URL points to a default photo. If the user hasn’t uploaded a photo and the request header included X-Connect-Theme: Salesforce1, this URL points to a default photo based on a theme that the admin selected for the org.</td>
<td>28.0</td>
</tr>
<tr>
<td>standardEmailPhotoUrl</td>
<td>String</td>
<td>A temporary URL to the small profile. The URL expires after 30 days and is available to unauthenticated users.</td>
<td>28.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>A resource that returns a Photo object: for example, /services/data/v59.0/chatter/users/005D0000001LL8OIAW/photo</td>
<td>28.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.ChatterGroup
- ConnectApi.RecommendationDefinition
- ConnectApi.User

**ConnectApi.PicklistRecordField**

Record field containing an enumerated value.
Subclass of ConnectApi.LabeledRecordField.

**ConnectApi.PinCapability**

If a feed element has this capability, users who have permission can pin it to a feed.
Subclass of ConnectApi.FeedElementCapability.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isPinnableByMe</td>
<td>Boolean</td>
<td>Specifies whether the context user can pin or unpin the entity to the feed (true) or not (false).</td>
<td>41.0</td>
</tr>
<tr>
<td>isPinned</td>
<td>Boolean</td>
<td>Specifies whether the entity is pinned (true) or not pinned (false) to the feed.</td>
<td>41.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.FeedElementCapabilities
ConnectApi.PinnedFeedElements

List of pinned feed elements for a feed.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>elements</td>
<td>List&lt;ConnectApi.FeedElement&gt;</td>
<td>List of pinned feed elements.</td>
<td>41.0</td>
</tr>
</tbody>
</table>

**Note:** In the UI, pinned feed elements don’t show all auxiliary information, such as comments, likes, interaction counts, or read by information. As a result, the ConnectApi.PinnedFeedElements output class doesn’t include all the information for these capabilities.

ConnectApi.PlatformAction

A platform action instance with state information for the context user.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>actionUrl</td>
<td>String</td>
<td>For action links of subtype Ui or Download, direct the user to download or visit the UI from this link. Salesforce issues a Javascript redirect for the link in this format: /action-link-redirect/communityId/actionLinkId?_bearer=bearerToken. For Api action links and for all platform actions, this value is null and Salesforce handles the call.</td>
<td>33.0</td>
</tr>
<tr>
<td>apiName</td>
<td>String</td>
<td>The API name. The value may be null.</td>
<td>33.0</td>
</tr>
<tr>
<td>confirmationMessage</td>
<td>String</td>
<td>If this action requires a confirmation and has a status of NewStatus, this is a default localized message that should be shown to an end user prior to invoking the action. Otherwise, this is null.</td>
<td>33.0</td>
</tr>
<tr>
<td>executingUser</td>
<td>ConnectApi.UserSummary</td>
<td>The user who initiated execution of this platform action.</td>
<td>33.0</td>
</tr>
<tr>
<td>groupDefault</td>
<td>Boolean</td>
<td>true if this platform action is the default or primary platform action in the platform action group; false otherwise. There can be only one default platform action per platform action group.</td>
<td>33.0</td>
</tr>
<tr>
<td>iconUrl</td>
<td>String</td>
<td>The URL of the icon for the platform action. This value may be null.</td>
<td>33.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>--------------------</td>
<td>---------------</td>
<td>-------------------------------------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>The ID for the platform action. If the type is <code>QuickAction</code> and the subtype is <code>Create</code>, this value is <code>null</code>.</td>
<td>33.0</td>
</tr>
<tr>
<td>label</td>
<td>String</td>
<td>The localized label for this platform action.</td>
<td>33.0</td>
</tr>
<tr>
<td>modifiedDate</td>
<td>Datetime</td>
<td>ISO 8601 format date string, for example, <code>2011-02-25T18:24:31.000Z</code>.</td>
<td>33.0</td>
</tr>
<tr>
<td>platformActionGroup</td>
<td>ConnectApi.Reference</td>
<td>A reference to the platform action group containing this platform action.</td>
<td>33.0</td>
</tr>
<tr>
<td>status</td>
<td>ConnectApi.PlatformActionStatus</td>
<td>The execution status of the platform action. Values are:&lt;br&gt;  • FailedStatus—The action link execution failed.&lt;br&gt;  • NewStatus—The action link is ready to be executed. Available for <code>Download</code> and <code>Ui</code> action links only.&lt;br&gt;  • PendingStatus—The action link is executing. Choosing this value triggers the API call for <code>Api</code> and <code>ApiAsync</code> action links.&lt;br&gt;  • SuccessfulStatus—The action link executed successfully.</td>
<td>33.0</td>
</tr>
<tr>
<td>subtype</td>
<td>String</td>
<td>The subtype of a platform action or <code>null</code>. If the type property is <code>ActionLink</code>, possible values are:&lt;br&gt;  • Api—The action link calls a synchronous API at the action URL. Salesforce sets the status to <code>SuccessfulStatus</code> or <code>FailedStatus</code> based on the HTTP status code returned by your server.&lt;br&gt;  • ApiAsync—The action link calls an asynchronous API at the action URL. The action remains in a <code>PendingStatus</code> state until a third party makes a request to <code>/connect/action-links/actionLinkId</code> to set the status to <code>SuccessfulStatus</code> or <code>FailedStatus</code> when the asynchronous operation is complete.&lt;br&gt;  • Download—The action link downloads a file from the action URL.&lt;br&gt;  • Ui—The action link takes the user to a web page at the action URL.</td>
<td>33.0</td>
</tr>
</tbody>
</table>
Note: Invoking \texttt{ApiAsync} action links from an app requires a call to set the status. However, there isn't currently a way to set the status of an action link using Apex. To set the status, use Connect REST API. See the Action Link resource in the Connect REST API Developer Guide for more information.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
</table>
| \texttt{type} | \texttt{ConnectApi\_PlatformActionType} | The type of platform action. Values are:  
  - \texttt{ActionLink}—An indicator on a feed element that targets an API, a web page, or a file, represented by a button in the Salesforce UI.  
  - \texttt{CustomButton}—When clicked, opens a URL or a Visualforce page in a window or executes JavaScript.  
  - \texttt{ProductivityAction}—Productivity actions are predefined and attached to a limited set of objects. Productivity actions include Send Email, Call, Map, View Website, and Read News. Except for the Call action, you can't edit productivity actions.  
  - \texttt{QuickAction}—A global or object-specific action.  
  - \texttt{StandardButton}—A predefined Salesforce button such as New, Edit, or Delete. | 33.0 |
| \texttt{url} | \texttt{String} | The URL for this platform action.  
If the type is \texttt{QuickAction} and the \texttt{subtype} is Create, this value is null. | 33.0 |

**SEE ALSO:**

\texttt{ConnectApi\_PlatformActionGroup}

**ConnectApi\_PlatformActionGroup**

A platform action group instance with state appropriate for the context user.

Action link groups are one type of platform action group and are therefore represented as \texttt{ConnectApi\_PlatformActionGroup} output classes.
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
</table>
| category      | ConnectApi.PlatformAction.GroupCategory | Indicates the priority and relative locations of platform actions. Values are:  
  - **Primary**—The action link group is displayed in the body of the feed element.  
  - **Overflow**—The action link group is displayed in the overflow menu of the feed element. | 33.0 |
| id            | String | The 18-character ID or an opaque string ID of the platform action group.  
  * If the ConnectApi.PlatformAction type is QuickAction and the subtype is Create, this value is null. | 33.0 |
| modifiedDate  | Datetime | ISO 8601 date string, for example, 2014-02-25T18:24:31.000Z. | 33.0 |
| platformActions | List<ConnectApi.PlatformAction> | The platform action instances for this group.  
  * Within an action link group, action links are displayed in the order listed in the actionLinks property of the ConnectApi.ActionLinkGroup DefinitionInput class.  
  * Within a feed item, action link groups are displayed in the order specified in the actionLinkGroupIds property of the ConnectApi.AssociatedActions CapabilityInput class. | 33.0 |
| url           | String | The URL for this platform action group.  
  * If the ConnectApi.PlatformAction type is QuickAction and the subtype is Create, this value is null. | 33.0 |

SEE ALSO:  
  - ConnectApi.AbstractRecommendation  
  - ConnectApi.AssociatedActionsCapability

**ConnectApi.PollCapability**

If a feed element has this capability, it includes a poll.  
Subclass of ConnectApi.FeedElementCapability.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>choices</td>
<td>List&lt;ConnectApi.FeedPollChoice&gt;</td>
<td>Collection of poll choices that make up the poll.</td>
<td>32.0</td>
</tr>
</tbody>
</table>
**Property Name** | **Type** | **Description** | **Available Version**
--- | --- | --- | ---
myChoiceId | String | 18-character ID of the poll choice that the context user has voted for in this poll. Returns null if the context user has not voted. | 32.0

totalVoteCount | Integer | Total number of votes cast on the feed poll element. | 32.0

SEE ALSO:
[ConnectApi.FeedElementCapabilities](#)

### ConnectApi.PostAuthorizationResponse

Gateway response following a post authorization request.

**Property Name** | **Type** | **Description** | **Available Version**
--- | --- | --- | ---
error | ConnectApi.ErrorResponse | Information about errors that occurred in the payment gateway while evaluating the post authorization request. | 54.0

gatewayResponse | ConnectApi.PostAuthGatewayResponse | Payment gateway’s response to the post authorization request. | 54.0

paymentAuthorization | ConnectApi.PaymentAuthorizationResponse | Payment gateway’s response to the original payment authorization request. | 54.0

paymentGatewayLogs | List.CONNECTApi.GatewayLogResponse> | Stores information exchanged between the Salesforce payments platform and external payment gateways. Gateway logs can also record payloads from external payment entities. | 54.0

paymentGroup | ConnectApi.PaymentGroupResponse | Payment group, consisting of one or more payments, sent to the gateway for the post authorization request. | 54.0

paymentMethod | ConnectApi.PaymentMethodResponse | Payment method used in the post authorization request. | 54.0

### ConnectApi.PostAuthGatewayResponse

Gateway response after confirmation that the merchant is ready to capture payment of an existing pre-authorized transaction. Subclass of [ConnectApi.AbstractGatewayResponse](#).
### ConnectApi.PreviewCancelOutputRepresentation

Expected financial values for a proposed cancel action.
Subclass of ConnectApi.BaseOutputRepresentation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>changeBalances</td>
<td>ConnectApi.ChangeItemOutputRepresentation</td>
<td>Expected financial values for the proposed cancel action.</td>
<td>48.0</td>
</tr>
<tr>
<td>orderSummaryId</td>
<td>String</td>
<td>ID of the OrderSummary.</td>
<td>48.0</td>
</tr>
</tbody>
</table>

### ConnectApi.PreviewReturnOutputRepresentation

Expected financial values for a proposed return action.
Subclass of ConnectApi.BaseOutputRepresentation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>changeBalances</td>
<td>ConnectApi.ChangeItemOutputRepresentation</td>
<td>Expected financial values for the proposed return action.</td>
<td>48.0</td>
</tr>
<tr>
<td>orderSummaryId</td>
<td>String</td>
<td>ID of the OrderSummary.</td>
<td>48.0</td>
</tr>
</tbody>
</table>

### ConnectApi.PriceAdjustmentSchedule

Price adjustment schedule.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>adjustmentMethod</td>
<td>String</td>
<td>Reserved for future use.</td>
<td>59.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the price adjustment schedule.</td>
<td>49.0</td>
</tr>
<tr>
<td>priceAdjustmentTiers</td>
<td>List&lt;ConnectApi.PriceAdjustmentTier&gt;</td>
<td>List of price adjustment tiers.</td>
<td>49.0</td>
</tr>
</tbody>
</table>
## ConnectApi.PriceAdjustmentTier

Price adjustment tier.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>adjustmentType</td>
<td>ConnectApi.PriceAdjustmentTierType</td>
<td>Type of price adjustment for the tier. Values are:</td>
<td>49.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- AmountBasedAdjustment—Price is adjusted by a specified amount.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- PercentageBasedAdjustment—Price is adjusted by a specified percentage.</td>
<td></td>
</tr>
<tr>
<td>adjustmentValue</td>
<td>String</td>
<td>Adjustment value of the tier.</td>
<td>49.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the price adjustment tier.</td>
<td>49.0</td>
</tr>
<tr>
<td>lowerBound</td>
<td>String</td>
<td>Lower limit of the tier.</td>
<td>49.0</td>
</tr>
<tr>
<td>tierUnitPrice</td>
<td>String</td>
<td>Unit price of the tier.</td>
<td>49.0</td>
</tr>
<tr>
<td>upperBound</td>
<td>String</td>
<td>Upper limit of the tier.</td>
<td>49.0</td>
</tr>
</tbody>
</table>

## ConnectApi.PricingResult

Product pricing result.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Three-letter ISO 4217 currency code associated with the product.</td>
<td>49.0</td>
</tr>
<tr>
<td>error</td>
<td>ConnectApi.ErrorResponse</td>
<td>Error code and message.</td>
<td>49.0</td>
</tr>
<tr>
<td>pricingLineItemLists</td>
<td>List&lt;ConnectApi.PricingResultLineItem&gt;</td>
<td>Pricing result line items.</td>
<td>49.0</td>
</tr>
<tr>
<td>success</td>
<td>Boolean</td>
<td>Specifies whether the execution was successful (true) or not (false).</td>
<td>49.0</td>
</tr>
</tbody>
</table>
### ConnectApi.PricingResultLineItem

Pricing result line item.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>error</td>
<td>ConnectApi.ErrorResponse</td>
<td>Error code and message.</td>
<td>49.0</td>
</tr>
<tr>
<td>listPrice</td>
<td>String</td>
<td>List price for the product.</td>
<td>49.0</td>
</tr>
<tr>
<td>lowestUnitPrice</td>
<td>String</td>
<td>Lowest unit price for the product.</td>
<td>49.0</td>
</tr>
<tr>
<td>pricebookEntryId</td>
<td>String</td>
<td>ID of the pricebook entry.</td>
<td>49.0</td>
</tr>
<tr>
<td>productId</td>
<td>String</td>
<td>ID of the product to price.</td>
<td>49.0</td>
</tr>
<tr>
<td>success</td>
<td>Boolean</td>
<td>Specifies whether the execution was successful (true) or not (false).</td>
<td>49.0</td>
</tr>
<tr>
<td>unitPrice</td>
<td>String</td>
<td>Unit price for the product.</td>
<td>49.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- [ConnectApi.PricingResult](#)
- [ConnectApi.ProductSummary](#)

### ConnectApi.ProductAttributeInfo

Product attribute metadata.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>allowableValues</td>
<td>List&lt;String&gt;</td>
<td>Active attribute picklist values that can be used to create variations. These values are determined by the order of the picklist values in Object Manager.</td>
<td>50.0</td>
</tr>
<tr>
<td>apiName</td>
<td>String</td>
<td>API name of the attribute.</td>
<td>50.0</td>
</tr>
<tr>
<td>availableValues</td>
<td>List&lt;String&gt;</td>
<td>Attribute picklist values that are available for the product in the store. These values are sorted by the order of values in the allowableValues property.</td>
<td>50.0</td>
</tr>
<tr>
<td>fieldEnumOrId</td>
<td>String</td>
<td>Field ID for custom fields or enumeration value for standard fields.</td>
<td>50.0</td>
</tr>
<tr>
<td>label</td>
<td>String</td>
<td>Label of the attribute.</td>
<td>50.0</td>
</tr>
<tr>
<td>objectName</td>
<td>String</td>
<td>Name of the object that contains the field.</td>
<td>50.0</td>
</tr>
</tbody>
</table>
### ConnectApi.ProductDetail

Product attribute.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>apiName</td>
<td>String</td>
<td>API name of the attribute.</td>
<td>50.0</td>
</tr>
<tr>
<td>label</td>
<td>String</td>
<td>Label of the attribute.</td>
<td>50.0</td>
</tr>
<tr>
<td>sequence</td>
<td>Integer</td>
<td>Sequence value determined by the order of the attributes under Commerce Setup for the attribute set.</td>
<td>50.0</td>
</tr>
<tr>
<td>value</td>
<td>String</td>
<td>Display value of the attribute.</td>
<td>50.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.ProductAttributesToProductEntry

### ConnectApi.ProductAttributeSelectionInfo

Product attribute.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>apiName</td>
<td>String</td>
<td>API name of the attribute.</td>
<td>50.0</td>
</tr>
<tr>
<td>label</td>
<td>String</td>
<td>Label of the attribute.</td>
<td>50.0</td>
</tr>
<tr>
<td>sequence</td>
<td>Integer</td>
<td>Sequence value determined by the order of the attributes under Commerce Setup for the attribute set.</td>
<td>50.0</td>
</tr>
<tr>
<td>value</td>
<td>String</td>
<td>Display value of the attribute.</td>
<td>50.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.ProductAttributesToProductEntry

### ConnectApi.ProductAttributeSet

Product attribute set data.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>attributes</td>
<td>Map&lt;String, String&gt;</td>
<td>Map of the attributes that are members of the attribute set.</td>
<td>50.0</td>
</tr>
<tr>
<td>developerName</td>
<td>String</td>
<td>Name of the attribute set.</td>
<td>50.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the product attribute record that represents the product attribute set.</td>
<td>50.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ProductAttributeSetInfo

Attribute set metadata.
Important: Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>attributeInfo</td>
<td>Map&lt;String, ConnectApi.ProductAttributeInfo&gt;</td>
<td>Map of the API name of the attribute field to the attribute metadata.</td>
<td>50.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the attribute set.</td>
<td>50.0</td>
</tr>
<tr>
<td>developerName</td>
<td>String</td>
<td>Developer name of the attribute set.</td>
<td>50.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the attribute set.</td>
<td>50.0</td>
</tr>
<tr>
<td>masterLabel</td>
<td>String</td>
<td>Label of the attribute set.</td>
<td>50.0</td>
</tr>
<tr>
<td>sequence</td>
<td>Integer</td>
<td>Sequence of the attribute set for the product.</td>
<td>50.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.ProductDetail

**ConnectApi.ProductAttributeSetSummary**

Summary of a product attribute set.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>apiName</td>
<td>String</td>
<td>API name of the attribute set.</td>
<td>51.0</td>
</tr>
<tr>
<td>attributes</td>
<td>List&lt;ConnectApi.ProductAttributeSummary&gt;</td>
<td>List of attributes in the attribute set.</td>
<td>51.0</td>
</tr>
<tr>
<td>label</td>
<td>String</td>
<td>Display label of the attribute set.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.OrderItemSummaryProduct
- ConnectApi.ProductSummary

**ConnectApi.ProductAttributeSummary**

Summary of a product attribute.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>apiName</td>
<td>String</td>
<td>API name of the attribute.</td>
<td>51.0</td>
</tr>
<tr>
<td>label</td>
<td>String</td>
<td>Display label of the attribute.</td>
<td>51.0</td>
</tr>
<tr>
<td>sequence</td>
<td>Integer</td>
<td>Sequence of the attribute in the attribute set.</td>
<td>51.0</td>
</tr>
</tbody>
</table>
### ConnectApi.ProductAttributeSetSummary

Mapping of an attribute value combination to a variation product ID.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>canonicalKey</td>
<td>String</td>
<td>Attribute API values concatenated with an underscore (_) based on the sequence number of the attributes in the attribute set.</td>
<td>50.0</td>
</tr>
<tr>
<td>productId</td>
<td>String</td>
<td>Variation product ID for the selection of attributes.</td>
<td>50.0</td>
</tr>
<tr>
<td>selectedAttributes</td>
<td>List&lt;ConnectApi.ProductAttributeSelectionInfo&gt;</td>
<td>Ordered list of attribute values and metadata that can be used to form a key that maps to product ID.</td>
<td>50.0</td>
</tr>
<tr>
<td>urlSlug</td>
<td>String</td>
<td>Variant URL slug for the selection of attributes.</td>
<td>59.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ProductCategoryData

Product category.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the category.</td>
<td>49.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the category.</td>
<td>49.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the category.</td>
<td>49.0</td>
</tr>
<tr>
<td>urlSlug</td>
<td>String</td>
<td>SEO-friendly URL slug of the category.</td>
<td>59.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ProductAttributesToProductEntry

Mapping of an attribute value combination to a variation product ID.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>value</td>
<td>String</td>
<td>Display value of the attribute.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ProductCategoryPath

### ConnectApi.SearchCategory
ConnectApi.ProductCategoryDetail

Details of a product category.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>bannerImage</td>
<td>ConnectApi.ProductCategoryMedia</td>
<td>Banner image of the product category.</td>
<td>49.0</td>
</tr>
<tr>
<td>fields</td>
<td>Map&lt;String, String&gt;</td>
<td>List of fields for the product category.</td>
<td>49.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the product category.</td>
<td>49.0</td>
</tr>
<tr>
<td>mediaGroups</td>
<td>List&lt;ConnectApi.ProductCategoryMediaGroup&gt;</td>
<td>List of media groups of the product category.</td>
<td>49.0</td>
</tr>
<tr>
<td>tileImage</td>
<td>ConnectApi.ProductCategoryMedia</td>
<td>Tile image of the product category.</td>
<td>49.0</td>
</tr>
<tr>
<td>urlSlug</td>
<td>String</td>
<td>SEO-friendly URL slug of the product category.</td>
<td>59.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.ProductCategoryDetailCollection

ConnectApi.ProductCategoryDetailCollection

Collection of product category details.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>productCategories</td>
<td>List&lt;ConnectApi.ProductCategoryDetail&gt;</td>
<td>List of product category details.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

ConnectApi.ProductCategoryMedia

Media associated with a product category.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>alternateText</td>
<td>String</td>
<td>Alternative text for the product category media.</td>
<td>49.0</td>
</tr>
<tr>
<td>contentVersionId</td>
<td>String</td>
<td>ID of the latest published content version if the media is stored as a ContentDocument. If the image is a customer-provided external URL, the value is null. Not supported in enhanced CMS workspaces.</td>
<td>49.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the product category image.</td>
<td>49.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>-----------------</td>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>mediaType</td>
<td><code>ConnectApi.ProductMediaType</code></td>
<td>Type of product media. Values are:</td>
<td>49.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Document</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Image</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Video</td>
<td></td>
</tr>
<tr>
<td>sortOrder</td>
<td><code>Integer</code></td>
<td>Sort order of a media item inside a media group.</td>
<td>49.0</td>
</tr>
<tr>
<td>thumbnailUrl</td>
<td><code>String</code></td>
<td>URL of the thumbnail for product media. If a value exists, it should be used</td>
<td>49.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>for the thumbnail whether the image is natively uploaded or hosted externally.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Not supported in enhanced CMS workspaces.</td>
<td></td>
</tr>
<tr>
<td>title</td>
<td><code>String</code></td>
<td>Title of the product category media.</td>
<td>49.0</td>
</tr>
<tr>
<td>url</td>
<td><code>String</code></td>
<td>URL of the product category media.</td>
<td>49.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- `ConnectApi.ProductCategoryMediaGroup`
- `ConnectApi.ProductCategoryDetail`

### ConnectApi.ProductCategoryMediaGroup

Media group associated with a product category.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>developerName</td>
<td><code>String</code></td>
<td>API name of the product category media group.</td>
<td>49.0</td>
</tr>
<tr>
<td>id</td>
<td><code>String</code></td>
<td>ID of the product category media group.</td>
<td>49.0</td>
</tr>
<tr>
<td>mediaItems</td>
<td><code>List&lt;ConnectApi.ProductCategoryMedia&gt;</code></td>
<td>List of media items within a product category media group.</td>
<td>49.0</td>
</tr>
<tr>
<td>name</td>
<td><code>String</code></td>
<td>Name of the product category media group.</td>
<td>49.0</td>
</tr>
<tr>
<td>usageType</td>
<td><code>ConnectApi.ProductMediaUsageType</code></td>
<td>Usage type of a product media item within a media group. Values are:</td>
<td>49.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Attachment—Product media group with product documents as attachments.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Banner—Product category media group with banner images of the product.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Listing—Product media group with listing images of the product.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Standard—Product media group with standard images and videos of the product.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Tile—Product category media group with tile images of the product.</td>
<td></td>
</tr>
</tbody>
</table>
**ConnectApi.ProductCategoryPath**

List of product categories in a path.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>path</td>
<td>List&lt;ConnectApi. ProductCategoryData&gt;</td>
<td>List of product categories in a path.</td>
<td>49.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.ProductDetail

**ConnectApi.ProductChild**

Child product related to a parent product.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>defaultQuantity</td>
<td>String</td>
<td>Default quantity of child products to be ordered.</td>
<td>57.0</td>
</tr>
<tr>
<td>productInfo</td>
<td>ConnectApi. ProductDetail</td>
<td>Product details of the child product.</td>
<td>57.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ProductChildCollection**

Collection of child products related to a parent product.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>count</td>
<td>Integer</td>
<td>Number of child products returned on this page.</td>
<td>57.0</td>
</tr>
<tr>
<td>currentPageToken</td>
<td>String</td>
<td>Current page token, if any.</td>
<td>57.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>URL of the current page in the response.</td>
<td>57.0</td>
</tr>
<tr>
<td>items</td>
<td>List&lt;ConnectApi. ProductChild&gt;</td>
<td>List of child products related to the parent product. The child products are sorted by their configured sequence values, in ascending order, with null values sorted last. If there are no configured sequence values, the child products are sorted by createdDate, in ascending order.</td>
<td>57.0</td>
</tr>
<tr>
<td>nextPageToken</td>
<td>String</td>
<td>Token for the next page, if any. A value is included in the response only if a value is returned for nextPageUrl.</td>
<td>57.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>URL of the next page, if any.</td>
<td>57.0</td>
</tr>
<tr>
<td>previousPageToken</td>
<td>String</td>
<td>Token for the previous page, if any. A value is included in the response only if a value is returned for previousPageUrl.</td>
<td>57.0</td>
</tr>
</tbody>
</table>
### Available Version

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>URL of the previous page, if any.</td>
<td>57.0</td>
</tr>
<tr>
<td>total</td>
<td>Integer</td>
<td>Total number of child products in the collection.</td>
<td>57.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ProductDetail

Details of a product.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>attributeSetInfo</td>
<td>Map&lt;String, ConnectApi.ProductAttributeSetInfo&gt;</td>
<td>Map of the attribute set developer name to its metadata.</td>
<td>50.0</td>
</tr>
<tr>
<td>defaultImage</td>
<td>ConnectApi.ProductMedia</td>
<td>Default image of the product.</td>
<td>49.0</td>
</tr>
<tr>
<td>entitlement</td>
<td>ConnectApi.ProductEntitlement</td>
<td>Entitlement details for the product. To get pricing information for products in version 57 and later, use the CommerceStorePricing Class.</td>
<td>49.0–56.0</td>
</tr>
<tr>
<td>fields</td>
<td>Map&lt;String, String&gt;</td>
<td>List of fields for the product.</td>
<td>49.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the product.</td>
<td>49.0</td>
</tr>
<tr>
<td>mediaGroups</td>
<td>List&lt;ConnectApi.ProductMediaGroup&gt;</td>
<td>List of media groups of the product.</td>
<td>49.0</td>
</tr>
<tr>
<td>primaryProductCategoryPath</td>
<td>ConnectApi.ProductCategoryPath</td>
<td>Primary category path of the product.</td>
<td>49.0</td>
</tr>
<tr>
<td>productClass</td>
<td>ConnectApi.ProductClass</td>
<td>Class of product. Values are:</td>
<td>50.0</td>
</tr>
<tr>
<td></td>
<td>• Simple</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Variation</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• VariationParent</td>
<td></td>
<td></td>
</tr>
<tr>
<td>productSellingModels</td>
<td>List&lt;ConnectApi.ProductSellingModel&gt;</td>
<td>List of product selling models for the product.</td>
<td>56.0</td>
</tr>
<tr>
<td>purchaseQuantityRule</td>
<td>ConnectApi.PurchaseQuantityRule</td>
<td>If one exists, purchase quantity rule for the product.</td>
<td>52.0</td>
</tr>
<tr>
<td>urlSlug</td>
<td>String</td>
<td>SEO-friendly URL slug for the product.</td>
<td>59.0</td>
</tr>
<tr>
<td>variationAttributeSet</td>
<td>ConnectApi.ProductAttributeSet</td>
<td>Variation attribute set for the product.</td>
<td>50.0</td>
</tr>
</tbody>
</table>
### ConnectApi.ProductVariationInfo

Available and allowable values for variation attributes and a map to resolve variation product IDs from attribute value combinations.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>variationInfo</td>
<td>ConnectApi.ProductVariationInfo</td>
<td></td>
<td>50.0</td>
</tr>
<tr>
<td>variationParentId</td>
<td>String</td>
<td>ID of the variation parent.</td>
<td>50.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ProductDetailsOutputRepresentation

Details about a product.

Subclass of ConnectApi.BaseOutputRepresentation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>attributes</td>
<td>List&lt;ConnectApi.ProductVariationAttributeOutputRepresentation&gt;</td>
<td>List of variation attributes that define variations of the product.</td>
<td>55.0</td>
</tr>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Currency ISO code.</td>
<td>55.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the product.</td>
<td>55.0</td>
</tr>
<tr>
<td>fields</td>
<td>Map&lt;String, String&gt;</td>
<td>List of the product's fields.</td>
<td>55.0</td>
</tr>
<tr>
<td>imageGroups</td>
<td>List&lt;ConnectApi.ProductImageGroupOutputRepresentation&gt;</td>
<td>List of the product's image groups.</td>
<td>55.0</td>
</tr>
<tr>
<td>listPrice</td>
<td>Double</td>
<td>List price.</td>
<td>55.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name.</td>
<td>55.0</td>
</tr>
<tr>
<td>productQuantityRule</td>
<td>Purchase Quantity Rule</td>
<td>If one exists, purchase quantity rule for the product.</td>
<td>55.0</td>
</tr>
<tr>
<td>productId</td>
<td>String</td>
<td>Product ID.</td>
<td>55.0</td>
</tr>
<tr>
<td>stockKeepingUnit</td>
<td>String</td>
<td>Stock keeping unit.</td>
<td>55.0</td>
</tr>
<tr>
<td>unitPrice</td>
<td>Double</td>
<td>Unit price.</td>
<td>55.0</td>
</tr>
<tr>
<td>variants</td>
<td>List&lt;ConnectApi.ProductVariantOutputRepresentation&gt;</td>
<td>List of variations of the product.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ProductEntitlement

Entitlements for a product.
Available Version | Description | Type | Property Name
--- | --- | --- | ---
49.0 | Specifies whether the product’s price can be viewed (true) or not (false). | Boolean | canViewPrice

SEE ALSO:  
ConnectApi.ProductDetail

ConnectApi.ProductImageOutputRepresentation

Details about a product image.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>alternateText</td>
<td>String</td>
<td>Alternate text for accessibility.</td>
<td>55.0</td>
</tr>
<tr>
<td>mediaType</td>
<td>String</td>
<td>Media type.</td>
<td>55.0</td>
</tr>
<tr>
<td>thumbnailUrl</td>
<td>String</td>
<td>URL of the thumbnail version of the product image.</td>
<td>55.0</td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>Title.</td>
<td>55.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>URL of the product image.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

ConnectApi.ProductImageGroupOutputRepresentation

Details about a product image group.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>images</td>
<td>List&lt;ConnectApi.ProductImageOutputRepresentation&gt;</td>
<td>List of product images in the group.</td>
<td>55.0</td>
</tr>
<tr>
<td>viewType</td>
<td>String</td>
<td>The type of product images in the group.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

ConnectApi.ProductMedia

Media associated with a product.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>alternateText</td>
<td>String</td>
<td>Alternative text for the product media.</td>
<td>49.0</td>
</tr>
<tr>
<td>contentVersionId</td>
<td>String</td>
<td>ID of the latest published content version if the media is stored as a ContentDocument. If the image is a customer-provided external URL, the value is null. Not supported in enhanced CMS workspaces.</td>
<td>49.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the product image.</td>
<td>49.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>mediaType</td>
<td><code>ConnectApi.ProductMediaType</code></td>
<td>Type of product media. Values are: Document, Image, Video</td>
<td>49.0</td>
</tr>
<tr>
<td>sortOrder</td>
<td>Integer</td>
<td>Sort order of a media item within a media group.</td>
<td>49.0</td>
</tr>
<tr>
<td>thumbnailUrl</td>
<td>String</td>
<td>URL of the thumbnail for product media. If a value exists, it should be used for the thumbnail whether the image is natively uploaded or hosted externally. Not supported in enhanced CMS workspaces.</td>
<td>49.0</td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>Title of the product media.</td>
<td>49.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>URL of the product media.</td>
<td>49.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- `ConnectApi.CartItemProduct`
- `ConnectApi.ProductDetail`
- `ConnectApi.ProductMediaGroup`
- `ConnectApi.OrderItemSummaryProduct`
- `ConnectApi.ProductSummary`

### `ConnectApi.ProductMediaGroup`

Media group associated with a product.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>developerName</td>
<td>String</td>
<td>API name of the product media group.</td>
<td>49.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the product media group.</td>
<td>49.0</td>
</tr>
<tr>
<td>mediaItems</td>
<td><code>List&lt;ConnectApi.ProductMedia&gt;</code></td>
<td>List of media items within a product media group.</td>
<td>49.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the product media group.</td>
<td>49.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>---------------</td>
<td>------</td>
<td>-------------</td>
<td>-------------------</td>
</tr>
</tbody>
</table>
| usageType     | ConnectApi.ProductMedia UsageType | Usage type of a product media item within a media group. Values are:  
- Attachment—Product media group with product documents as attachments.  
- Banner—Product category media group with banner images of the product.  
- Listing—Product media group with listing images of the product.  
- Standard—Product media group with standard images and videos of the product.  
- Tile—Product category media group with tile images of the product. | 49.0 |
| defaultImage  | ConnectApi.ProductMedia | Media representation of the product's default image. | 54.0 |
| error         | ConnectApi.ErrorResponse | Error code and error message. | 54.0 |
| fields        | Map<String, String> | Map of fields belonging to the product. | 54.0 |
| id            | String | ID of the product. | 54.0 |
| prices        | ConnectApi.PricingResult LineItem | Price of the product.  
To get pricing information for products in version 58 and later, use the CommerceStorePricing Class. | 54.0–57.0 |
| sku           | String | SKU of the product. | 54.0 |
| success       | Boolean | Represents whether execution was successful and product overview information was retrieved without error. | 54.0 |

**SEE ALSO:**  
ConnectApi.ProductDetail  
ConnectApi.ProductCategoryDetail

**ConnectApi.ProductOverview**

Overview of a product, with summary information about prices, selected fields, and the product’s default image.

**ConnectApi.ProductOverviewCollection**

Collection of product overviews.
### Available Version Description Type Property Name

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>products</td>
<td>List&lt;ConnectApi.ProductOverview&gt;</td>
<td>Collection of product overview.</td>
<td>54.0</td>
</tr>
<tr>
<td>total</td>
<td>Integer</td>
<td>Total number of products returned.</td>
<td>54.0</td>
</tr>
</tbody>
</table>

#### ConnectApi.ProductPrice

Pricing information for a product.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Three-letter ISO 4217 currency code associated with the product. Products are priced using the currency for the buyer account or guest buyer profile. If your store doesn’t support the currency for the buyer account or guest buyer profile, products are priced using the default currency for your store.</td>
<td>49.0</td>
</tr>
<tr>
<td>listPrice</td>
<td>String</td>
<td>List price for the product.</td>
<td>49.0</td>
</tr>
<tr>
<td>lowestUnitPrice</td>
<td>String</td>
<td>Lowest unit price for the product.</td>
<td>49.0</td>
</tr>
<tr>
<td>priceAdjustment</td>
<td>ConnectApi.PriceAdjustmentSchedule</td>
<td>Price adjustment schedule for the product. If a product selling model ID is specified in a request parameter, this property is empty.</td>
<td>49.0</td>
</tr>
<tr>
<td>pricebookEntryId</td>
<td>String</td>
<td>ID of the price book entry. If a product selling model ID is specified in a request parameter, this property is empty.</td>
<td>49.0</td>
</tr>
<tr>
<td>productPriceEntries</td>
<td>List&lt;ConnectApi.ProductPriceEntry&gt;</td>
<td>List of line item prices for the product.</td>
<td>56.0</td>
</tr>
<tr>
<td>unitPrice</td>
<td>String</td>
<td>Unit price for the product. If a product selling model ID is specified in a request parameter, this property is empty.</td>
<td>49.0</td>
</tr>
</tbody>
</table>

#### ConnectApi.ProductPriceEntry

Line item price for the product.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>error</td>
<td>ConnectApi.ErrorResponse</td>
<td>Error code and error message.</td>
<td>56.0</td>
</tr>
<tr>
<td>listPrice</td>
<td>String</td>
<td>List price for the product entry.</td>
<td>56.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>-------------------------</td>
<td>---------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>priceAdjustment</td>
<td>ConnectApi.PriceAdjustmentSchedule</td>
<td>Price adjustment schedule.</td>
<td>56.0</td>
</tr>
<tr>
<td>pricebookEntryId</td>
<td>String</td>
<td>ID of the pricebook entry.</td>
<td>56.0</td>
</tr>
<tr>
<td>productSellingModelId</td>
<td>String</td>
<td>ID of the product selling model. If no product selling model ID is specified</td>
<td>56.0</td>
</tr>
<tr>
<td>success</td>
<td>Boolean</td>
<td>Specifies whether execution was successful (true) or not (false).</td>
<td>56.0</td>
</tr>
<tr>
<td>unitPrice</td>
<td>String</td>
<td>Unit price for the product entry.</td>
<td>56.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ProductVariationInfo**

Product variation attributes, metadata, and mappings of attribute combinations to variation product IDs.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>attributesToProductMappings</td>
<td>List&lt;ConnectApi.ProductAttributesToProductEntry&gt;</td>
<td>List ordered by ProductAttribute.Sequence values that map the attribute value combinations to the variation product ID.</td>
<td>50.0</td>
</tr>
<tr>
<td>variationAttributeInfo</td>
<td>Map&lt;String, ConnectApi.ProductAttributeInfo&gt;</td>
<td>Map of field API name to product attribute information.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

**See Also:**

ConnectApi.ProductDetail

**ConnectApi.ProductSearchFacetOutputRepresentation**

Product search facet value.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>attributeType</td>
<td>String</td>
<td>Attribute type of the facet value.</td>
<td>59.0</td>
</tr>
<tr>
<td>displayName</td>
<td>String</td>
<td>Display name of the facet value.</td>
<td>59.0</td>
</tr>
<tr>
<td>displayRank</td>
<td>Integer</td>
<td>Display rank of the facet value.</td>
<td>59.0</td>
</tr>
<tr>
<td>displayType</td>
<td>String</td>
<td>Display type of the facet value.</td>
<td>59.0</td>
</tr>
<tr>
<td>facetType</td>
<td>String</td>
<td>Type of the facet.</td>
<td>59.0</td>
</tr>
<tr>
<td>nameOrId</td>
<td>String</td>
<td>Name or ID of the facet.</td>
<td>59.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>--------------------</td>
<td>---------------</td>
<td>-------------------------------------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>values</td>
<td><a href="https://example.com">ConnectApi.ProductSearchFacetValueOutputRepresentation</a></td>
<td>A list of facet values for the search.</td>
<td>59.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ProductSearchFacetValueOutputRepresentation**

Output representation of a product search facet value.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>displayName</td>
<td>String</td>
<td>Display name of the search facet.</td>
<td>59.0</td>
</tr>
<tr>
<td>nameOrId</td>
<td>String</td>
<td>Unique name or ID of the search facet.</td>
<td>59.0</td>
</tr>
<tr>
<td>productCount</td>
<td>Integer</td>
<td>Number of products found with the search facet.</td>
<td>59.0</td>
</tr>
<tr>
<td>type</td>
<td>String</td>
<td>Type of the search facet.</td>
<td>59.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ProductSearchImageOutputRepresentation**

Output representation of the product search image.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>alternateText</td>
<td>String</td>
<td>Alternate text for the product image.</td>
<td>59.0</td>
</tr>
<tr>
<td>mediaType</td>
<td>String</td>
<td>Media type of the product image.</td>
<td>59.0</td>
</tr>
<tr>
<td>sortOrder</td>
<td>Integer</td>
<td>Sort order of the product image.</td>
<td>59.0</td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>Title of the product image.</td>
<td>59.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>URL of the product image.</td>
<td>59.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ProductSearchOutputRepresentation**

Output representation of the product search response.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Three-letter ISO 4217 currency code associated with the product.</td>
<td>59.0</td>
</tr>
<tr>
<td>facets</td>
<td><a href="https://example.com">ConnectApi.ProductSearchFacetOutputRepresentation</a></td>
<td>A list of facet names to filter the search. For example, [&quot;size_medium&quot;, &quot;color_red&quot;]</td>
<td>59.0</td>
</tr>
<tr>
<td>locale</td>
<td>String</td>
<td>Locale used for the product search.</td>
<td>59.0</td>
</tr>
<tr>
<td>pageNumber</td>
<td>Integer</td>
<td>Maximum number of search results pages to return. If you don’t specify a value, the default is 1.</td>
<td>59.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------------------------------------</td>
<td>--------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>pageSize</td>
<td>Integer</td>
<td>Number of items per page. Valid values are from 1 through 100. If you don’t specify a value, the default size is 20.</td>
<td>59.0</td>
</tr>
<tr>
<td>products</td>
<td>List&lt;ConnectApi.ProductOutputRepresentation&gt;</td>
<td>List of products found by the search.</td>
<td>59.0</td>
</tr>
<tr>
<td>totalRecordsFound</td>
<td>Integer</td>
<td>Total products found.</td>
<td>59.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ProductSearchProductOutputRepresentation**

Product found by a product search.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the product.</td>
<td>59.0</td>
</tr>
<tr>
<td>image</td>
<td>ConnectApi.ProductImageOutputRepresentation</td>
<td>Image of the product.</td>
<td>59.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the product.</td>
<td>59.0</td>
</tr>
<tr>
<td>productClass</td>
<td>String</td>
<td>Class of the product.</td>
<td>59.0</td>
</tr>
<tr>
<td>stockKeepingUnit</td>
<td>String</td>
<td>Stock Keeping Unit (SKU) of the product.</td>
<td>59.0</td>
</tr>
<tr>
<td>variationAttributeSet</td>
<td>ConnectApi.ProductVariationAttributeOutputRepresentation</td>
<td>Variation attribute set of the product.</td>
<td>59.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ProductSearchResults**

Product search results.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>categories</td>
<td>ConnectApi.ProductSearchCategory</td>
<td>Categories from the search results.</td>
<td>52.0</td>
</tr>
<tr>
<td>correlationId</td>
<td>String</td>
<td>Reserved for future use.</td>
<td>55.0</td>
</tr>
<tr>
<td>facets</td>
<td>List&lt;ConnectApi.ProductSearchFacet&gt;</td>
<td>Facets from the search results.</td>
<td>52.0</td>
</tr>
<tr>
<td>locale</td>
<td>String</td>
<td>Locale of the search results.</td>
<td>52.0</td>
</tr>
<tr>
<td>productsPage</td>
<td>ConnectApi.ProductSummaryPage</td>
<td>Page of products from the search results.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ProductSearchSuggestionsResults**

Product search suggestions results.
### recentSearchSuggestions

Suggestions based on the user’s recent searches.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>recentSearchSuggestions</td>
<td><code>List&lt;ConnectApi.SearchSuggestion&gt;</code></td>
<td>Suggestions based on the user’s recent searches.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ProductSellingModel

Product selling model for Commerce subscriptions.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td><code>String</code></td>
<td>ID of the product selling model.</td>
<td>56.0</td>
</tr>
<tr>
<td>name</td>
<td><code>String</code></td>
<td>Name of the product selling model.</td>
<td>56.0</td>
</tr>
<tr>
<td>pricingTerm</td>
<td><code>Integer</code></td>
<td>Number of pricing term units in the pricing term. Used with <code>pricingTermUnit</code> to define the length of the pricing term. For example, if <code>pricingTermUnit</code> is <code>Months</code> and this property is 1, the subscription is priced monthly. However, if the <code>sellingModelType</code> property is set to <code>OneTime</code>, the <code>pricingTerm</code> property is empty, because the product isn’t sold as a subscription. The only allowed value for this property is 1.</td>
<td>56.0</td>
</tr>
<tr>
<td>pricingTermUnit</td>
<td><code>ConnectApi.PricingTermUnit</code></td>
<td>Unit of time used to define a pricing term. Value is:</td>
<td>56.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Months—Product is priced on a monthly basis.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Annual—Product is priced on an annual basis.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>This unit of time is combined with a number (specified by the <code>pricingTerm</code> property) to define the full term of the subscription. For example, if the unit of time is <code>Months</code> and the <code>pricingTerm</code> property is set to 1, the subscription is priced monthly. However, if the <code>sellingModelType</code> property is set to <code>OneTime</code>, the <code>pricingTermUnit</code> property is empty, because the product isn’t sold as a subscription.</td>
<td></td>
</tr>
<tr>
<td>sellingModelType</td>
<td><code>ConnectApi.SellingModelType</code></td>
<td>Type of product selling model. Values are:</td>
<td>56.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Evergreen—A subscription without an end date. An evergreen subscription continues until the shopper affirmatively cancels it.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• OneTime—A product that isn’t sold as a subscription.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• TermDefined—A subscription with a defined end date. The subscription continues for a</td>
<td></td>
</tr>
</tbody>
</table>
When the term ends, the subscription ends.

Rules for the subscription term.

### ConnectApi.ProductSummary

Product summary.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>defaultImage</td>
<td>ConnectApi.ProductMedia</td>
<td>Default image of the product.</td>
<td>52.0</td>
</tr>
<tr>
<td>fields</td>
<td>Map&lt;String, ConnectApi.FieldValue&gt;</td>
<td>Map of fields belonging to the product.</td>
<td>52.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the product.</td>
<td>52.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the product.</td>
<td>52.0</td>
</tr>
<tr>
<td>prices</td>
<td>ConnectApi.PricingResult LineItem</td>
<td>Prices of the product.</td>
<td>52.0</td>
</tr>
<tr>
<td>productClass</td>
<td>ConnectApi.ProductClass</td>
<td>Class of product. Values are:</td>
<td>52.0</td>
</tr>
<tr>
<td></td>
<td>• Simple</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Variation</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• VariationParent</td>
<td></td>
<td></td>
</tr>
<tr>
<td>productSellingModelInformation</td>
<td>ConnectApi.CommerceProductSellingModel</td>
<td>Product selling model information.</td>
<td>59.0</td>
</tr>
<tr>
<td>purchaseQuantityRule</td>
<td>ConnectApi.PurchaseQuantityRule</td>
<td>If one exists, purchase quantity rule for the product.</td>
<td>52.0</td>
</tr>
<tr>
<td>variationAttributeSet</td>
<td>ConnectApi.ProductAttributeSetSummary</td>
<td>Variation attribute set that’s associated with the product.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

ConnectApi.ProductSummaryPage
## ConnectApi.ProductSummaryPage

Page of product summaries.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Three-letter ISO 4217 currency code associated with the product.</td>
<td>52.0</td>
</tr>
<tr>
<td>pageSize</td>
<td>Integer</td>
<td>Number of products per page in the search results.</td>
<td>52.0</td>
</tr>
<tr>
<td>products</td>
<td>List&lt;String&gt;</td>
<td>Collection of product summaries.</td>
<td>52.0</td>
</tr>
<tr>
<td>total</td>
<td>Long</td>
<td>Total number of products in the search results.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.ProductSearchResults

## ConnectApi.ProductVariantOutputRepresentation

Details about a product variation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>listPrice</td>
<td>Double</td>
<td>List price.</td>
<td>55.0</td>
</tr>
<tr>
<td>price</td>
<td>Double</td>
<td>Price.</td>
<td>55.0</td>
</tr>
<tr>
<td>productId</td>
<td>String</td>
<td>Product ID.</td>
<td>55.0</td>
</tr>
<tr>
<td>stockKeepingUnit</td>
<td>String</td>
<td>Stock Keeping Unit.</td>
<td>55.0</td>
</tr>
<tr>
<td>unitPrice</td>
<td>Double</td>
<td>Unit price.</td>
<td>55.0</td>
</tr>
<tr>
<td>variationValues</td>
<td>Map&lt;String, String&gt;</td>
<td>The variation attribute values that define the variation.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

## ConnectApi.ProductVariationAttributeOutputRepresentation

Details about a product variation attribute.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>apiName</td>
<td>String</td>
<td>API name of the attribute.</td>
<td>55.0</td>
</tr>
<tr>
<td>label</td>
<td>String</td>
<td>Label of the attribute.</td>
<td>55.0</td>
</tr>
<tr>
<td>variationAttributeValues</td>
<td>List&lt;String&gt;</td>
<td>List of valid values for the variation attribute.</td>
<td>55.0</td>
</tr>
</tbody>
</table>
**ConnectApi.ProductVariationAttributeValueOutputRepresentation**
Valid value for a product variation attribute.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>API Name of the attribute this value belongs to.</td>
<td>55.0</td>
</tr>
<tr>
<td>orderable</td>
<td>Boolean</td>
<td>Whether the value defines an orderable product variation.</td>
<td>55.0</td>
</tr>
<tr>
<td>value</td>
<td>String</td>
<td>Value of the value.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

**ConnectApi.PromotionCart**
A cart, its items, and its adjustment groups.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>cartAdjustmentGroups</td>
<td>ConnectApi.PromotionCartAdjustmentGroup</td>
<td>Cart adjustment groups belonging to the cart.</td>
<td>57.0</td>
</tr>
<tr>
<td>cartItems</td>
<td>ConnectApi.PromotionCartItem</td>
<td>Cart items belonging to the cart.</td>
<td>57.0</td>
</tr>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Currency code of the cart.</td>
<td>57.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the cart.</td>
<td>57.0</td>
</tr>
<tr>
<td>totalAdjustmentBaseAmount</td>
<td>String</td>
<td>Total adjustment base amount for the cart.</td>
<td>57.0</td>
</tr>
<tr>
<td>totalNetAmount</td>
<td>String</td>
<td>Total price of the cart, including adjustments.</td>
<td>57.0</td>
</tr>
<tr>
<td>totalProductBaseAmount</td>
<td>String</td>
<td>Total price of all cart items in the cart.</td>
<td>57.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.PromotionEvaluation
evaluate(salesTransaction)

**ConnectApi.PromotionCartAdjustmentGroup**
Adjustment group associated with a cart.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>adjustmentBasisReference</td>
<td>String</td>
<td>ID of the associated coupon, if applicable.</td>
<td>57.0</td>
</tr>
<tr>
<td>adjustmentDescription</td>
<td>String</td>
<td>Description of the adjustment.</td>
<td>57.0</td>
</tr>
</tbody>
</table>
### Available Version Description Type Property Name

<table>
<thead>
<tr>
<th>adjustmentType</th>
<th>description</th>
<th>string</th>
<th>ConnectApi.AdjustmentType</th>
<th>57.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>adjustmentValue</td>
<td>price value of the adjustment.</td>
<td>string</td>
<td>57.0</td>
<td></td>
</tr>
<tr>
<td>baseAmount</td>
<td>total amount of the adjustment.</td>
<td>string</td>
<td>57.0</td>
<td></td>
</tr>
<tr>
<td>cartId</td>
<td>ID of the cart.</td>
<td>string</td>
<td>57.0</td>
<td></td>
</tr>
<tr>
<td>id</td>
<td>ID of the cart adjustment group.</td>
<td>string</td>
<td>57.0</td>
<td></td>
</tr>
<tr>
<td>priceAdjustment</td>
<td>ID of the related promotion.</td>
<td>string</td>
<td>57.0</td>
<td></td>
</tr>
<tr>
<td>priority</td>
<td>where in the sequence of adjustments this adjustment was applied.</td>
<td>integer</td>
<td>57.0</td>
<td></td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.PromotionCart
- ConnectApi.PromotionEvaluation
evaluate(salesTransaction)

### ConnectApi.PromotionCartItem

A cart item and its adjustments.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>cartDeliveryGroupId</td>
<td>string</td>
<td>ID of the associated cart delivery group.</td>
<td>57.0</td>
</tr>
<tr>
<td>cartId</td>
<td>string</td>
<td>ID of the associated cart.</td>
<td>57.0</td>
</tr>
<tr>
<td>cartItemPriceAdjustments</td>
<td>string</td>
<td>List of price adjustments applied to the cart item.</td>
<td>57.0</td>
</tr>
<tr>
<td>id</td>
<td>string</td>
<td>ID of the cart item.</td>
<td>57.0</td>
</tr>
<tr>
<td>itemDescription</td>
<td>string</td>
<td>Description of the cart item.</td>
<td>57.0</td>
</tr>
<tr>
<td>itemName</td>
<td>string</td>
<td>Name of the cart item.</td>
<td>57.0</td>
</tr>
<tr>
<td>listPrice</td>
<td>string</td>
<td>Unit list price of the cart item.</td>
<td>57.0</td>
</tr>
<tr>
<td>product2Id</td>
<td>string</td>
<td>ID of the product.</td>
<td>57.0</td>
</tr>
<tr>
<td>quantity</td>
<td>string</td>
<td>Quantity of the cart item.</td>
<td>57.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------------------------</td>
<td>------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>salesPrice</td>
<td>String</td>
<td>Unit sales price of the cart item.</td>
<td>57.0</td>
</tr>
<tr>
<td>sku</td>
<td>String</td>
<td>Stock keeping unit of the cart item.</td>
<td>57.0</td>
</tr>
<tr>
<td>totalAdjustment</td>
<td>String</td>
<td>Total adjustment amount for the cart item.</td>
<td>57.0</td>
</tr>
<tr>
<td>BaseAmount</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>totalLine</td>
<td>String</td>
<td>Total amount for the cart item, based on sales price and quantity, not including adjustments.</td>
<td>57.0</td>
</tr>
<tr>
<td>BaseAmount</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>totalList</td>
<td>String</td>
<td>Total amount for the cart item, based on list price and quantity, not including adjustments.</td>
<td>57.0</td>
</tr>
<tr>
<td>BaseAmount</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>totalNetAmount</td>
<td>String</td>
<td>Total amount for the cart item, based on list price and quantity, including adjustments.</td>
<td>57.0</td>
</tr>
<tr>
<td>type</td>
<td>ConnectApi.CartItemType</td>
<td>Type of item in a cart. Values are:</td>
<td>57.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• DeliveryCharge</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Product</td>
<td></td>
</tr>
</tbody>
</table>

**SEE ALSO:**

- ConnectApi.PromotionCart
- ConnectApi.PromotionEvaluation
- evaluate(salesTransaction)

### ConnectApi.PromotionCartItemPriceAdjustment

Price adjustments applied to a cart item.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>adjustmentAmount</td>
<td>ConnectApi.AdjustmentAmountScope</td>
<td>Scope of the price adjustment amount. Values are:</td>
<td>57.0</td>
</tr>
<tr>
<td>Scope</td>
<td></td>
<td>• Total—The adjustment scope is the total price.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Unit—The adjustment scope is the unit price.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• UnproratedTotal—The adjustment scope is the unprorated total price.</td>
<td></td>
</tr>
<tr>
<td>adjustmentBasis</td>
<td>String</td>
<td>ID of the associated coupon, if applicable.</td>
<td>57.0</td>
</tr>
<tr>
<td>Reference</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>adjustmentDescription</td>
<td>String</td>
<td>Description of the adjustment.</td>
<td>57.0</td>
</tr>
<tr>
<td>adjustmentTarget</td>
<td>ConnectApi.CartPromotionType</td>
<td>Level of the promotion target. Values are:</td>
<td>57.0</td>
</tr>
<tr>
<td>Type</td>
<td></td>
<td>• Cart—The target is cart-level.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Item—The target is item-level.</td>
<td></td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>------------------------</td>
<td>--------------------------</td>
<td>-------------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>adjustmentType</td>
<td>ConnectApi.AdjustmentType</td>
<td>How the price adjustment amount is calculated. Values are:</td>
<td>57.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AdjustmentAmount—The adjustment is a fixed amount.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• AdjustmentPercentage—The adjustment is a percentage.</td>
<td></td>
</tr>
<tr>
<td>adjustmentValue</td>
<td>String</td>
<td>Value of the price adjustment.</td>
<td>57.0</td>
</tr>
<tr>
<td>baseAmount</td>
<td>String</td>
<td>Total adjustment amount.</td>
<td>57.0</td>
</tr>
<tr>
<td>cartAdjustmentGroupId</td>
<td>String</td>
<td>ID of the associated cart adjustment group.</td>
<td>57.0</td>
</tr>
<tr>
<td>cartItemId</td>
<td>String</td>
<td>ID of the associated cart item.</td>
<td>57.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the cart item price adjustment.</td>
<td>57.0</td>
</tr>
<tr>
<td>priceAdjustmentCauseId</td>
<td>String</td>
<td>ID of the associated promotion.</td>
<td>57.0</td>
</tr>
<tr>
<td>priority</td>
<td>Integer</td>
<td>Where in the sequence of adjustments this adjustment was applied.</td>
<td>57.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.PromotionCartItem
ConnectApi.PromotionCart
ConnectApi.PromotionEvaluation
evaluate(salesTransaction)

**ConnectApi.PromotionCoupon**
A coupon used in a promotion.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>couponCode</td>
<td>String</td>
<td>Coupon code</td>
<td>57.0</td>
</tr>
<tr>
<td>couponErrorCode</td>
<td>String</td>
<td>Error code returned if the coupon is invalid.</td>
<td>57.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the coupon</td>
<td>57.0</td>
</tr>
<tr>
<td>isValidCoupon</td>
<td>Boolean</td>
<td>Indicates whether the coupon is valid (true) or invalid (false).</td>
<td>57.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.PromotionEvaluation
evaluate(salesTransaction)
### ConnectApi.PromotionEvaluation

Results of a promotion evaluation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>cart</td>
<td>ConnectApi.PromotionCart</td>
<td>Cart and its items.</td>
<td>57.0</td>
</tr>
<tr>
<td>coupons</td>
<td>List&lt;ConnectApi.PromotionCoupon&gt;</td>
<td>List of coupon codes to enable promotions. A customer can apply a maximum of two coupons.</td>
<td>57.0</td>
</tr>
</tbody>
</table>

SEE ALSO: 
evaluate(salesTransaction)

### ConnectApi.PurchaseQuantityRule

Rule that restricts the quantity of a product that can be purchased.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>increment</td>
<td>String</td>
<td>Increment value of the quantity that can be purchased.</td>
<td>52.0</td>
</tr>
<tr>
<td>maximum</td>
<td>String</td>
<td>Maximum quantity that can be purchased.</td>
<td>52.0</td>
</tr>
<tr>
<td>minimum</td>
<td>String</td>
<td>Minimum quantity that can be purchased.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

SEE ALSO: 
ConnectApi.CartItemProduct
ConnectApi.ProductDetail
ConnectApi.ProductSummary

### ConnectApi.QuestionAndAnswersCapability

If a feed element has this capability, it has a question and comments on the feed element are answers to the question.
Subclass of ConnectApi.FeedElementCapability.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>bestAnswer</td>
<td>ConnectApi.Comment</td>
<td>Comment selected as the best answer for the question.</td>
<td>32.0</td>
</tr>
<tr>
<td>bestAnswerSelectedBy</td>
<td>ConnectApi.UserSummary</td>
<td>User who selected the best answer for the question.</td>
<td>32.0</td>
</tr>
<tr>
<td>canCurrentUserSelectOrRemoveBestAnswer</td>
<td>Boolean</td>
<td>Indicates whether the context user can select or remove a best answer (true) or not (false).</td>
<td>32.0</td>
</tr>
<tr>
<td>candidateAnswers</td>
<td>ConnectApi.CandidateAnswersStatus</td>
<td>Status of candidate answers for the question.</td>
<td>41.0</td>
</tr>
</tbody>
</table>
### Available Version Description Type Property Name

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>escalatedCase</td>
<td>ConnectApi. Reference</td>
<td>If a question post is escalated, this is the case to which it was escalated.</td>
<td>33.0</td>
</tr>
<tr>
<td>questionTitle</td>
<td>String</td>
<td>Title for the question.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

- [ConnectApi.FeedElementCapabilities](#)
- [ConnectApi.QuestionAndAnswersSuggestions](#)
- [ConnectApi.RankAverageDistanceOutputRepresentation](#)

#### ConnectApi.QuestionAndAnswersSuggestions

Question and answers suggestions.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>articles</td>
<td>List&lt;ConnectApi. ArticleItem&gt;</td>
<td>List of articles.</td>
<td>32.0</td>
</tr>
<tr>
<td>questions</td>
<td>List&lt;ConnectApi. FeedElement&gt;</td>
<td>List of questions.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

#### ConnectApi.RankAverageDistanceOutputRepresentation

The results of calculating the average distances from sets of inventory locations to an order recipient.

Subclass of [ConnectApi.BaseOutputRepresentation](#).

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>distanceUnit</td>
<td>String</td>
<td>The specified unit of distance (miles or kilometers).</td>
<td>51.0</td>
</tr>
<tr>
<td>results</td>
<td>List&lt;ConnectApi. AverageDistanceResult OutputRepresentation&gt;</td>
<td>The results of the shipping distance calculations.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

#### ConnectApi.ReadBy

Information about who read the feed element and when.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>lastReadDateByUser</td>
<td>Datetime</td>
<td>When the user last read the feed element in ISO 8601 format.</td>
<td>40.0</td>
</tr>
</tbody>
</table>
**ConnectApi.ReadByCapability**

If a feed element has this capability, the context user can mark it as read.

Subclass of `ConnectApi.FeedElementCapability`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isReadByMe</td>
<td>Boolean</td>
<td>Specifies whether the feed element has been read (true) or not (false) by the context user.</td>
<td>40.0</td>
</tr>
<tr>
<td>lastReadDateByMe</td>
<td>Datetime</td>
<td>Last date when the feed element was marked read for the context user in ISO 8601 format. Otherwise, null.</td>
<td>40.0</td>
</tr>
<tr>
<td>page</td>
<td><code>ConnectApi.ReadByPage</code></td>
<td>First page of information about who read the feed element and when.</td>
<td>40.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- `ConnectApi.FeedElementCapabilities`

**ConnectApi.ReadByPage**

A collection of information about who read the feed element and when.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageToken</td>
<td>String</td>
<td>Token identifying the current page.</td>
<td>40.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page. The default is 25 items per page.</td>
<td>40.0</td>
</tr>
<tr>
<td>items</td>
<td><code>List&lt;ConnectApi.ReadBy&gt;</code></td>
<td>Collection of read-by information, including users and when they last read the feed element.</td>
<td>40.0</td>
</tr>
<tr>
<td>nextPageToken</td>
<td>String</td>
<td>Token identifying the next page, or <code>null</code> if there isn't a next page.</td>
<td>40.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or <code>null</code> if there isn't a next page.</td>
<td>40.0</td>
</tr>
<tr>
<td>previousPageToken</td>
<td>String</td>
<td>Reserved for future use.</td>
<td>40.0</td>
</tr>
</tbody>
</table>
### ConnectApi.Recommendation

A Next Best Action recommendation object.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>acceptanceLabel</td>
<td>String</td>
<td>Text indicating user acceptance of the recommendation.</td>
<td>45.0</td>
</tr>
<tr>
<td>actionReference</td>
<td>String</td>
<td>Reference to the action to perform, for example, launching a flow.</td>
<td>45.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the recommendation.</td>
<td>45.0</td>
</tr>
<tr>
<td>externalId</td>
<td>String</td>
<td>External ID of the recommendation. This ID doesn’t need to be a Salesforce 18-character ID. For example, it can be a product number from an external system.</td>
<td>46.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the recommendation.</td>
<td>45.0</td>
</tr>
<tr>
<td>image</td>
<td>ConnectApi.FileAsset</td>
<td>Image to display.</td>
<td>45.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the recommendation.</td>
<td>45.0</td>
</tr>
<tr>
<td>rejectionLabel</td>
<td>String</td>
<td>Text indicating user rejection of the recommendation.</td>
<td>45.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>URL to the recommendation.</td>
<td>45.0</td>
</tr>
</tbody>
</table>

### ConnectApi.RecommendationAudience

A custom recommendation audience.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>criteria</td>
<td>ConnectApi.AudienceCriteria</td>
<td>The criteria for the custom recommendation audience type.</td>
<td>36.0</td>
</tr>
</tbody>
</table>

### SEE ALSO:

- ConnectApi.ReadByCapability
### Available Version

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>18-character ID of the custom recommendation audience.</td>
<td>35.0</td>
</tr>
<tr>
<td>memberCount</td>
<td>Integer</td>
<td>Number of members in the custom recommendation audience.</td>
<td>35.0 only</td>
</tr>
<tr>
<td>members</td>
<td></td>
<td>Members of the custom recommendation audience.</td>
<td>35.0 only</td>
</tr>
<tr>
<td>modifiedBy</td>
<td></td>
<td>User who last modified the custom recommendation audience.</td>
<td>36.0</td>
</tr>
<tr>
<td>modifiedDate</td>
<td>Datetime</td>
<td>ISO 8601 format date string, for example, 2011-02-25T18:24:31.000Z.</td>
<td>36.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the custom recommendation audience.</td>
<td>35.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>URL to the custom recommendation audience.</td>
<td>35.0</td>
</tr>
</tbody>
</table>

### Available Version

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>audienceCount</td>
<td>Integer</td>
<td>The total number of custom recommendation audiences.</td>
<td>35.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>URL to the current page.</td>
<td>35.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>URL to the next page.</td>
<td>35.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>URL to the previous page.</td>
<td>35.0</td>
</tr>
<tr>
<td>recommendation Audiences</td>
<td>List&lt;ConnectApi.Recommendation Audience&gt;</td>
<td>A list of custom recommendation audiences.</td>
<td>35.0</td>
</tr>
</tbody>
</table>
**ConnectApi.RecommendationCollection**

A list of Chatter, custom, and static recommendations.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>recommendations</td>
<td>List&lt;ConnectApi.AbstractRecommendation&gt;</td>
<td>Collection of Chatter, custom, and static recommendations.</td>
<td>33.0</td>
</tr>
</tbody>
</table>

**ConnectApi.RecommendationDefinition**

Represents a custom recommendation definition.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>actionUrl</td>
<td>String</td>
<td>The URL for acting on this custom recommendation.</td>
<td>35.0</td>
</tr>
<tr>
<td>actionUrlName</td>
<td>String</td>
<td>The text label for the action URL in the user interface.</td>
<td>35.0</td>
</tr>
<tr>
<td>explanation</td>
<td>String</td>
<td>Explanation of the custom recommendation definition.</td>
<td>35.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>18-character ID of the custom recommendation definition.</td>
<td>35.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the custom recommendation definition. The name is displayed in Setup.</td>
<td>35.0</td>
</tr>
<tr>
<td>photo</td>
<td>ConnectApi.Photo</td>
<td>Photo of the custom recommendation definition.</td>
<td>35.0</td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>Title of the custom recommendation definition.</td>
<td>35.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>URL to the Connect REST API resource for the custom recommendation definition.</td>
<td>35.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.RecommendationDefinitionPage
- ConnectApi.ScheduledRecommendation

**ConnectApi.RecommendationDefinitionPage**

A list of custom recommendation definitions.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>recommendation Definitions</td>
<td>List&lt;ConnectApi.Recommendation Definition&gt;</td>
<td>A list of custom recommendation definitions.</td>
<td>35.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>URL to the Connect REST API resource for the recommendation definition collection.</td>
<td>35.0</td>
</tr>
</tbody>
</table>
**ConnectApi.RecommendationExplanation**

Explanation for a Chatter recommendation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>detailsUrl</td>
<td>String</td>
<td>URL to explanation details or null if the Chatter recommendation doesn’t have a detailed explanation.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
ConnectApi.AbstractRecommendation

**ConnectApi.RecommendationReaction**

A reaction to a recommendation produced by a recommendation strategy

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>aiModel</td>
<td>String</td>
<td>Reserved for future use.</td>
<td>47.0</td>
</tr>
<tr>
<td>contextRecord</td>
<td>ConnectApi.Reference</td>
<td>Reference to the context record.</td>
<td>45.0</td>
</tr>
<tr>
<td>createdBy</td>
<td>ConnectApi.Reference</td>
<td>Reference to the reaction creator.</td>
<td>45.0</td>
</tr>
<tr>
<td>createdDate</td>
<td>Datetime</td>
<td>Reaction creation date.</td>
<td>45.0</td>
</tr>
<tr>
<td>externalId</td>
<td>String</td>
<td>External target ID of the recommendation reacted on. This ID doesn’t need to be a Salesforce 18-character ID. For example, it can be a product number from an external system.</td>
<td>46.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>Reaction record ID.</td>
<td>45.0</td>
</tr>
<tr>
<td>onBehalfOf</td>
<td>ConnectApi.Reference</td>
<td>Reference to the user or record that is indirectly reacting to the recommendation.</td>
<td>45.0</td>
</tr>
<tr>
<td>reactionType</td>
<td>ConnectApi.RecommendationReactionType</td>
<td>Type of reaction to a recommendation. Values are:</td>
<td>45.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Accepted</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Rejected</td>
<td></td>
</tr>
<tr>
<td>recommendation Mode</td>
<td>String</td>
<td>Reserved for future use.</td>
<td>46.0</td>
</tr>
<tr>
<td>recommendation Score</td>
<td>Double</td>
<td>Reserved for future use.</td>
<td>46.0</td>
</tr>
<tr>
<td>strategy</td>
<td>ConnectApi.RecordSnapshot</td>
<td>Strategy that recommended the target record.</td>
<td>45.0</td>
</tr>
</tbody>
</table>
### ConnectApi.RecommendedObject

A recommended object, such as a custom or static recommendation.

Subclass of ConnectApi.Actor

### ConnectApi.RecommendationReactions

A list of recommendation reactions.

### ConnectApi.RecommendationsCapability

If a feed element has this capability, it has a recommendation.

Subclass of ConnectApi.FeedElementCapability.
### ConnectApi.RecordCapability

If a comment has this capability, it has a record attachment.
Subclass of `ConnectApi.FeedElementCapability`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>recordId</td>
<td>String</td>
<td>ID of the record.</td>
<td>42.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>URL to the record.</td>
<td>42.0</td>
</tr>
</tbody>
</table>

### ConnectApi.RecordField

Generic record field containing a label and text value.
Subclass of `ConnectApi.LabeledRecordField`.
No additional properties.

SEE ALSO:
- `ConnectApi.CompoundRecordField`
- `ConnectApi.OrderItemSummary`
- `ConnectApi.OrderItemSummaryProduct`
- `ConnectApi.OrderDeliveryGroupSummary`
- `ConnectApi.OrderSummaryRepresentation`

### ConnectApi.RecordSnapshot

A record snapshot in a recommendation reaction.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the record.</td>
<td>45.0</td>
</tr>
<tr>
<td>nameAtSnapshot</td>
<td>String</td>
<td>Name of the record when the ID was recorded.</td>
<td>45.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- `ConnectApi.RecommendationReaction`
**ConnectApi.RecordSnapshotCapability**

If a feed element has this capability, it contains all the snapshotted fields of a record for a single create record event.

Subclass of `ConnectApi.FeedElementCapability`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>recordView</td>
<td><code>ConnectApi.RecordView</code></td>
<td>A record representation that includes metadata and data so you can display the record easily.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

SEE ALSO:  
`ConnectApi.FeedElementCapabilities`

**ConnectApi.RecordSummary**

Record summary.

Subclass of `ConnectApi.AbstractRecordView`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>entityLabel</td>
<td><code>ConnectApi.EntityLabel</code></td>
<td>Label of the record’s entity.</td>
<td>40.0</td>
</tr>
</tbody>
</table>

SEE ALSO:  
`ConnectApi.EmailAddress`  
`ConnectApi.EmailAttachment`  
`ConnectApi.ReferenceRecordField`  
`ConnectApi.ReferenceWithDateRecordField`

**ConnectApi.RecordSummaryList**

Summary information about a list of records in the organization including custom objects.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>records</td>
<td><code>List&lt;ConnectApi.ActorWithId&gt;</code></td>
<td>A list of records.</td>
<td>30.0</td>
</tr>
<tr>
<td>url</td>
<td><code>String</code></td>
<td>The URL to this list of records.</td>
<td>30.0</td>
</tr>
</tbody>
</table>

**ConnectApi.RecordView**

A view of any record in the org, including a custom object record. This object is used if a specialized object, such as User or ChatterGroup, isn’t available for the record type.

Subclass of `ConnectApi.AbstractRecordView`.

1940
### ConnectApi.RecordViewSection

Section of record fields and values on a record detail.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>sections</td>
<td>List&lt;ConnectApi.RecordViewSection&gt;</td>
<td>List of record view sections.</td>
<td>29.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.RecordSnapshotCapability

### ConnectApi.RecordsetFilterCriteriaCollection

List of the recordset filters and records.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>recordsetFilters</td>
<td>List&lt;ConnectApi.RecordsetFilterCriteria&gt;</td>
<td>Collection of recordset filter criteria IDs and filtered record IDs.</td>
<td>53.0</td>
</tr>
</tbody>
</table>

### ConnectApi.RecordsetFilterCriteria

Recordset filter criteria and the filtered records.
### ConnectApi.RecordsetFilterCriteriaCollection

Reference to a record.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>The ID of the record being referenced, which could be an 18-character ID or some other string identifier.</td>
<td>28.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>The URL to the resource endpoint.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ReferenceRecordField

Record field with a label and text value.
Subclass of ConnectApi.LabeledRecordField.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>reference</td>
<td>ConnectApi.RecordSummary</td>
<td>Object referenced by the record field.</td>
<td>29.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ReferencedRefundResponse

Refund comprehensive output.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>error</td>
<td>ConnectApi.ErrorResponse</td>
<td>Error response representation for the refund.</td>
<td>50.0</td>
</tr>
<tr>
<td>gatewayResponse</td>
<td>ConnectApi.RefundGatewayResponse</td>
<td>Gateway response received for the processed refund request.</td>
<td>50.0</td>
</tr>
<tr>
<td>paymentGatewayLogs</td>
<td>List&lt;ConnectApi.GatewayLogResponse&gt;</td>
<td>Gateway log collection representation for the refund.</td>
<td>50.0</td>
</tr>
<tr>
<td>paymentGroup</td>
<td>ConnectApi.PaymentGroupResponse</td>
<td>Payment group associated with the refund.</td>
<td>50.0</td>
</tr>
<tr>
<td>refund</td>
<td>ConnectApi.RefundResponse</td>
<td>Refund response representation.</td>
<td>50.0</td>
</tr>
</tbody>
</table>
**ConnectApi.ReferenceWithDateRecordField**

Record field containing a referenced object that acted at a specific time, for example, "Created By...".
Subclass of `ConnectApi.LabeledRecordField`.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>dateValue</td>
<td>Datetime</td>
<td>Time at which the referenced object acted.</td>
<td>29.0</td>
</tr>
<tr>
<td>reference</td>
<td><code>ConnectApi.RecordSummary</code></td>
<td>Object referenced by the record field.</td>
<td>29.0</td>
</tr>
</tbody>
</table>

**ConnectApi.RefundGatewayResponse**

Refund gateway response.
No additional properties.

**ConnectApi.RefundResponse**

Refund output.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>accountId</td>
<td>String</td>
<td>ID of the account related to the refund record.</td>
<td>50.0</td>
</tr>
<tr>
<td>amount</td>
<td>Double</td>
<td>Total amount of the refund transaction performed in the payment request.</td>
<td>50.0</td>
</tr>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Three-letter ISO 4217 currency code associated with the payment group record.</td>
<td>50.0</td>
</tr>
<tr>
<td>effectiveDate</td>
<td>Datetime</td>
<td>Date that the refund becomes effective.</td>
<td>50.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the refund record.</td>
<td>50.0</td>
</tr>
<tr>
<td>refundNumber</td>
<td>String</td>
<td>Number of the refund record that was created as a result of the request processing.</td>
<td>50.0</td>
</tr>
<tr>
<td>requestDate</td>
<td>Datetime</td>
<td>Date when the refund occurred.</td>
<td>50.0</td>
</tr>
<tr>
<td>status</td>
<td>String</td>
<td>Indicates the results of processing the refund transaction in the gateway. Can be DRAFT, PROCESSED or CANCELLED.</td>
<td>50.0</td>
</tr>
</tbody>
</table>

**ConnectApi.RelatedFeedPost**

This class is abstract.
Subclass of `ConnectApi.ActorWithId`.
Superclass of `ConnectApi.RelatedQuestion`.

1943
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>score</td>
<td>Double</td>
<td>Score of the related feed post that indicates how closely related it is</td>
<td>37.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>to the context feed post.</td>
<td></td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>Title of the related feed post.</td>
<td>37.0</td>
</tr>
</tbody>
</table>

SEE ALSO: ConnectApi.RelatedFeedPosts

**ConnectApi.RelatedFeedPosts**

A collection of related feed posts.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>relatedFeedPosts</td>
<td>List&lt;ConnectApi.RelatedFeedPost&gt;</td>
<td>Collection of related feed posts.</td>
<td>37.0</td>
</tr>
</tbody>
</table>

**ConnectApi.RelatedQuestion**

A related question.
Subclass of ConnectApi.RelatedFeedPost.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>hasBestAnswer</td>
<td>Boolean</td>
<td>Indicates whether the question has a best answer.</td>
<td>37.0</td>
</tr>
<tr>
<td>interactions</td>
<td>ConnectApi.InteractionsCapability</td>
<td>The number of individual views, likes, and comments on a question.</td>
<td>38.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ReleaseHeldFOCapacityOutputRepresentation**

Response to a request to confirm held fulfillment order capacity at one or more locations. Can correspond to one action call.
Subclass of ConnectApi.BaseOutputRepresentation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>releaseHeldFOCapacityResponses</td>
<td>List&lt;ConnectApi.ReleaseHeldFOCapacityResponseOutputRepresentation&gt;</td>
<td>List of responses to the requests to confirm held fulfillment order capacity at one or more locations.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ReleaseHeldFOCapacityResponseOutputRepresentation**

Response to a request to release held fulfillment order capacity at one or more locations.
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>capacityResponses</td>
<td>List&lt;ConnectApi.CapacityResponse&gt;</td>
<td>List of responses to the requests to release held fulfillment order capacity at individual locations.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ReplyIntent**

Reply intent for a social post.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>managedSocialAccount</td>
<td>ConnectApi.ManagedSocialAccount</td>
<td>Managed social account that replies to the social post.</td>
<td>45.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

ConnectApi.ReplyIntents

**ConnectApi.ReplyIntents**

List of reply intents for a social post.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>replies</td>
<td>List&lt;ConnectApi.ReplyIntent&gt;</td>
<td>List of reply intents for the social post.</td>
<td>45.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

ConnectApi.SocialPostIntents

**ConnectApi.RepositoryFileDetail**

A detailed description of a repository file.
Subclass of ConnectApi.AbstractRepositoryFile.
No additional properties.

**ConnectApi.RepositoryFileSummary**

A summary of a repository file.
Subclass of ConnectApi.AbstractRepositoryFile.
No additional properties.

SEE ALSO:

ConnectApi.RepositoryFolderItem
ConnectApi.RepositoryFolderDetail
A detailed description of a repository folder.
Subclass of ConnectApi.AbstractRepositoryFolder.
No additional properties.

ConnectApi.RepositoryFolderItem
A folder item.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>file</td>
<td>ConnectApi. Repository FileSummary</td>
<td>If the folder item is a file, the file summary. If the folder item is a folder, null.</td>
<td>39.0</td>
</tr>
<tr>
<td>folder</td>
<td>ConnectApi. Repository FolderSummary</td>
<td>If the folder item is a folder, the folder summary. If the folder item is a file, null.</td>
<td>39.0</td>
</tr>
</tbody>
</table>
| type | ConnectApi. FolderItemType | Type of item in a folder. Values are:  
  • file  
  • folder | 39.0 |

SEE ALSO:
  ConnectApi.RepositoryFolderItemsCollection

ConnectApi.RepositoryFolderItemsCollection
A collection of repository folder items.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>URL to the current page of items.</td>
<td>39.0</td>
</tr>
<tr>
<td>items</td>
<td>List&lt;ConnectApi. Repository FolderItem&gt;</td>
<td>Collection of items in a repository folder.</td>
<td>39.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>URL to the next page of items, or null if there isn’t a next page.</td>
<td>39.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>URL to the previous page of items, or null if there isn’t a previous page.</td>
<td>39.0</td>
</tr>
</tbody>
</table>

ConnectApi.RepositoryFolderSummary
A summary of a repository folder.
Subclass of `ConnectApi.AbstractRepositoryFolder`.
No additional properties.

SEE ALSO:
`ConnectApi.RepositoryFolderItem`

**ConnectApi.RepositoryGroupSummary**

A group summary.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
</table>
| groupType     | `ConnectApi.ContentHub.GroupType` | Type of group. Values are:
                    • `Everybody`—Group is public to everybody.
                    • `EverybodyInDomain`—Group is public to everybody in the same domain.
                    • `Unknown`—Group type is unknown. | 39.0               |
| name          | `String`                  | Name of the group.                                                          | 39.0               |

SEE ALSO:
`ConnectApi.ExternalFilePermissionInformation`

**ConnectApi.RepositoryUserSummary**

A user summary.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>firstName</td>
<td><code>String</code></td>
<td>First name of the user.</td>
<td>39.0</td>
</tr>
<tr>
<td>lastName</td>
<td><code>String</code></td>
<td>Last name of the user.</td>
<td>39.0</td>
</tr>
</tbody>
</table>

**ConnectApi.Reputation**

Reputation for a user.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>reputationLevel</td>
<td><code>ConnectApi.ReputationLevel</code></td>
<td>User's reputation level.</td>
<td>32.0</td>
</tr>
<tr>
<td>reputationPoints</td>
<td><code>Double</code></td>
<td>User's reputation points, which can be earned by performing different activities.</td>
<td>32.0</td>
</tr>
</tbody>
</table>
**Available Version**\n**Description**\n**Type**\n**Property Name**

---

### ConnectApi.ReputationLevel

Reputation level for a user.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>levelImageUrl</td>
<td>String</td>
<td>URL to the reputation level image.</td>
<td>32.0</td>
</tr>
<tr>
<td>levelName</td>
<td>String</td>
<td>Name of the reputation level.</td>
<td>32.0</td>
</tr>
<tr>
<td>levelNumber</td>
<td>Integer</td>
<td>Reputation level number, which is the numerical rank of the level,</td>
<td>32.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>with the lowest level at 1. Administrators define the reputation level</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>point ranges.</td>
<td></td>
</tr>
</tbody>
</table>

**SEE ALSO:**

- [ConnectApi.Reputation](#)

---

### ConnectApi.RequestHeader

An HTTP request header name and value pair.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>The name of the request</td>
<td>33.0</td>
</tr>
<tr>
<td>value</td>
<td>String</td>
<td>header.</td>
<td>33.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

- [ConnectApi.ActionLinkDefinition](#)

---

### ConnectApi.ResourceLinkSegment

Resource link segment.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>url</td>
<td>String</td>
<td>URL to a resource not otherwise identified by an ID field, for example,</td>
<td>28.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>a link to a list of users.</td>
<td></td>
</tr>
</tbody>
</table>
**ConnectApi.ReturnItemsOutputRepresentation**

Output of Return Items. Includes the ID of the generated change order for items and delivery charges being returned, as well as the ID of the generated change order for any charged return fees. Also includes information about any ReturnOrderLineItems that were created to represent remaining return quantities.

Subclass of `ConnectApi.BaseOutputRepresentation`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>changeOrderId</td>
<td>String</td>
<td>ID of the change order created by processing the ReturnOrderLineItems...</td>
<td>52.0</td>
</tr>
<tr>
<td>feeChangeOrderId</td>
<td>String</td>
<td>ID of the change order created by processing the ReturnOrderLineItems...</td>
<td>56.0</td>
</tr>
<tr>
<td>returnLineItemSplits</td>
<td>List&lt;String&gt;</td>
<td>List of properties representing any remaining quantities from partial returns processed by this call. It includes order items, delivery charges, and return fees. Each element of the list includes the ID of a split ReturnOrderLineItem and the ID of the partially processed ReturnOrderLineItem whose remaining quantity it holds.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
```
returnItems(returnOrderId, returnItemsInput)
```

**ConnectApi.ReturnOrderLineItemSplitLineOutputRepresentation**

After a change order is created for a ReturnOrderLineItem, that ReturnOrderLineItem is read-only. If the Return Items API is used to return a partial quantity, it creates a new “split” ReturnOrderLineItem to hold the remaining quantity to be returned. In that case, it returns this output property, which contains the IDs of the original and split ReturnOrderLineItems.

Subclass of `ConnectApi.BaseOutputRepresentation`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>newReturnOrderItemId</td>
<td>String</td>
<td>ID of the new ReturnOrderLineItem that holds the remaining return quantity.</td>
<td>52.0</td>
</tr>
<tr>
<td>originalReturnOrderItemId</td>
<td>String</td>
<td>ID of the original ReturnOrderLineItem.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
```
ConnectApi.ReturnItemsOutputRepresentation
```
```
returnItems(returnOrderId, returnItemsInput)
```
ConnectApi.ReturnOrderOutputRepresentation

ID of the created ReturnOrder.
Subclass of ConnectApi.BaseOutputRepresentation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>returnOrderId</td>
<td>String</td>
<td>ID of the created ReturnOrder.</td>
<td>50.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
createReturnOrder(returnOrderInput)

ConnectApi.SaleResponse

Payment sale response.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>error</td>
<td>ConnectApi.ErrorResponse</td>
<td>Error representation for the payment sale.</td>
<td>54.0</td>
</tr>
<tr>
<td>gatewayResponse</td>
<td>ConnectApi.SaleGatewayResponse</td>
<td>Information from the payment gateway following the sale request.</td>
<td>54.0</td>
</tr>
<tr>
<td>payment</td>
<td>ConnectApi.PaymentResponse</td>
<td>Information about the payment used in the sale request.</td>
<td>54.0</td>
</tr>
<tr>
<td>paymentGatewayLogs</td>
<td>List&lt;ConnectApi.GatewayLogResponse&gt;</td>
<td>Collection of responses from the gateway following the sale request.</td>
<td>54.0</td>
</tr>
<tr>
<td>paymentGroup</td>
<td>ConnectApi.PaymentGroupResponse</td>
<td>Payment group used in the sale request.</td>
<td>54.0</td>
</tr>
<tr>
<td>paymentMethod</td>
<td>ConnectApi.PaymentMethodResponse</td>
<td>Payment method used in the sale request.</td>
<td>54.0</td>
</tr>
</tbody>
</table>

ConnectApi.SaleGatewayResponse

Sale gateway response.
No additional properties.

ConnectApi.ScheduledRecommendation

Represents a scheduled custom recommendation.
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
</table>
| channel       | ConnectApi.Recommendation Channel | A way to tie custom recommendations together. For example, display recommendations in specific places in the UI or show recommendations based on time of day or geographic locations. Values are:  
  • CustomChannel1—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels. For example, community managers can use Experience Builder to determine where recommendations appear.  
  • CustomChannel2—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.  
  • CustomChannel3—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.  
  • CustomChannel4—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.  
  • CustomChannel5—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.  
  • DefaultChannel—Default recommendation channel. Recommendations appear by default on the Home and Question Detail pages of Customer Service and Partner Central Experience Builder templates. They also appear in the feed in the Salesforce mobile web and anywhere community managers add recommendations using Experience Builder. | 36.0              |
<p>| enabled       | Boolean  | Indicates whether scheduling is enabled. If true, the custom recommendation is enabled and appears in Experience Cloud sites. If false, custom recommendations in feeds in Salesforce mobile web aren't removed, but no new custom recommendations appear. In Customer Service and Partner Central sites, disabled custom recommendations no longer appear. | 35.0              |
| id            | String   | 18-character ID of the scheduled custom recommendation.                                                                                                                                                     | 35.0              |</p>
<table>
<thead>
<tr>
<th><strong>Property Name</strong></th>
<th><strong>Type</strong></th>
<th><strong>Description</strong></th>
<th><strong>Available Version</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>rank</td>
<td>Integer</td>
<td>The rank determining the order of this scheduled custom recommendation.</td>
<td>35.0</td>
</tr>
<tr>
<td>recommendation AudienceId</td>
<td>String</td>
<td>ID of the audience for the scheduled custom recommendation.</td>
<td>35.0</td>
</tr>
<tr>
<td>recommendation Definition Representation</td>
<td>ConnectApi.RecommendationDefinition</td>
<td>Custom recommendation definition that this scheduled recommendation schedules.</td>
<td>35.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>URL to the Connect REST API resource for the scheduled custom recommendation.</td>
<td>35.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
ConnectApi.ScheduledRecommendationPage

**ConnectApi.ScheduledRecommendationPage**
A list of scheduled custom recommendations.

<table>
<thead>
<tr>
<th><strong>Property Name</strong></th>
<th><strong>Type</strong></th>
<th><strong>Description</strong></th>
<th><strong>Available Version</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>scheduled Recommendations</td>
<td>List&lt;ConnectApi.ScheduledRecommendation&gt;</td>
<td>A list of scheduled custom recommendations.</td>
<td>35.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>URL to the Connect REST API resource for the scheduled custom recommendation collection.</td>
<td>35.0</td>
</tr>
</tbody>
</table>

**ConnectApi.Scope**
Scope information for a target.

<table>
<thead>
<tr>
<th><strong>Property Name</strong></th>
<th><strong>Type</strong></th>
<th><strong>Description</strong></th>
<th><strong>Available Version</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the scope for the target.</td>
<td>48.0–49.0</td>
</tr>
<tr>
<td>value</td>
<td>String</td>
<td>Value of the scope for the target.</td>
<td>48.0–49.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
ConnectApi.Target

**ConnectApi.SearchCategory**
Search category.
### Available Version

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>category</td>
<td>ConnectApi.ProductCategoryData</td>
<td>Information about the category.</td>
<td>52.0</td>
</tr>
<tr>
<td>children</td>
<td>List&lt;ConnectApi.SearchCategory&gt;</td>
<td>First-level child categories of the category searched with non-empty search results.</td>
<td>52.0</td>
</tr>
<tr>
<td>productCount</td>
<td>Long</td>
<td>Number of products in the search results that belong to the category.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.ProductSearchResults

### ConnectApi.SearchFacet

Search facet.

This class is abstract and is a superclass of ConnectApi.DistinctValueSearchFacet.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>attributeType</td>
<td>ConnectApi.CommerceSearchAttributeType</td>
<td>Search attribute type. Values are:</td>
<td>52.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Custom</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ProductAttribute</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ProductCategory</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Product2</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Standard</td>
<td></td>
</tr>
<tr>
<td>displayName</td>
<td>String</td>
<td>Display name of the facet.</td>
<td>52.0</td>
</tr>
<tr>
<td>displayRank</td>
<td>Integer</td>
<td>Display rank of the facet. Valid values are from 1 through 50.</td>
<td>52.0</td>
</tr>
<tr>
<td>displayType</td>
<td>ConnectApi.CommerceSearchFacetDisplayType</td>
<td>Display type of the facet. Values are:</td>
<td>52.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• CategoryTree</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• DatePicker</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• MultiSelect</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• SingleSelect</td>
<td></td>
</tr>
<tr>
<td>facetType</td>
<td>ConnectApi.CommerceSearchFacetFacetType</td>
<td>Search facet type. Value is:</td>
<td>52.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• DistinctValue</td>
<td></td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>----------------</td>
<td>---------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>nameOrId</td>
<td>String</td>
<td>Developer name of the attribute. In version 52.0 and later, the ID of the attribute isn’t supported.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
*ConnectApi.ProductSearchResults*

**ConnectApi.SearchSuggestion**

Search suggestion.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>value</td>
<td>String</td>
<td>Search suggestion.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
*ConnectApi.ProductSearchSuggestionsResults*

**ConnectApi.ServiceAppointmentOutput**

Output of the create service appointment request.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>result</td>
<td>ConnectApi.ServiceAppointmentResult</td>
<td>Result of the create or update service appointment request.</td>
<td>53.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ServiceAppointmentResult**

Contains result of the service appointment.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>assignedResourceIds</td>
<td>List&lt;String&gt;</td>
<td>The IDs of the assigned resources.</td>
<td>53.0</td>
</tr>
<tr>
<td>parentRecordId</td>
<td>String</td>
<td>The ID of the parent record.</td>
<td>53.0</td>
</tr>
<tr>
<td>serviceAppointmentId</td>
<td>String</td>
<td>The ID of the service appointment record.</td>
<td>53.0</td>
</tr>
</tbody>
</table>

**ConnectApi.ShiftsFromPattern**

Shifts created from a pattern.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>count</td>
<td>Integer</td>
<td>Total count of created shifts.</td>
<td>51.0</td>
</tr>
</tbody>
</table>
### ConnectApi.ShiftsFromPatternError

Shifts from pattern error response.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>error</td>
<td>ConnectApi.ShiftsFromPatternError</td>
<td>Error details for shifts from a pattern.</td>
<td>53.0</td>
</tr>
<tr>
<td>isSuccess</td>
<td>Boolean</td>
<td>Indicates if the request is successful (true) or not (false).</td>
<td>53.0</td>
</tr>
<tr>
<td>recordIds</td>
<td>List&lt;String&gt;</td>
<td>Collection of created shift IDs.</td>
<td>51.0</td>
</tr>
</tbody>
</table>

### ConnectApi.SiteSearchItem

Site search result item.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>contentReference</td>
<td>String</td>
<td>Content reference field, which is the route developer name for a site page or a content key for a content detail page.</td>
<td>54.0</td>
</tr>
<tr>
<td>contentTypeDeveloperName</td>
<td>String</td>
<td>Developer name of the content type of the site search result item.</td>
<td>54.0</td>
</tr>
<tr>
<td>highlightedSnippet</td>
<td>String</td>
<td>Text snippet that contains the query term.</td>
<td>54.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the site search result item.</td>
<td>54.0</td>
</tr>
<tr>
<td>pageType</td>
<td>ConnectApi.SitesPageType</td>
<td>Type of site search result item. Values are:</td>
<td>54.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- ContentPage</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- SitePage</td>
<td></td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>Title of the site search result item.</td>
<td>54.0</td>
</tr>
</tbody>
</table>

SEE ALSO:  
ConnectApi.SiteSearchResult
## ConnectApi.SiteSearchResult

Site search result.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageToken</td>
<td>String</td>
<td>Token for the current page of search results.</td>
<td>54.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>URL to the current page of search results.</td>
<td>54.0</td>
</tr>
<tr>
<td>items</td>
<td>List</td>
<td>Collection of search result items.</td>
<td>54.0</td>
</tr>
<tr>
<td>language</td>
<td>String</td>
<td>Language of the search results.</td>
<td>54.0</td>
</tr>
<tr>
<td>nextPageToken</td>
<td>String</td>
<td>Token for the next page of search results.</td>
<td>54.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>URL to the next page of search results, or null if there isn’t a next page.</td>
<td>54.0</td>
</tr>
<tr>
<td>pageSize</td>
<td>Integer</td>
<td>Number of items per page in search results.</td>
<td>54.0</td>
</tr>
<tr>
<td>previousPageToken</td>
<td>String</td>
<td>Token for the previous page of search results.</td>
<td>54.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>URL to the previous page of search results, or null if there isn’t a previous page.</td>
<td>54.0</td>
</tr>
<tr>
<td>totalItems</td>
<td>Integer</td>
<td>Total number of items in the search results across all pages.</td>
<td>54.0</td>
</tr>
</tbody>
</table>

## ConnectApi.SocialAccount

A social account on a social network.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>externalSocialAccountId</td>
<td>String</td>
<td>ID of the external social account, if available.</td>
<td>38.0</td>
</tr>
<tr>
<td>handle</td>
<td>String</td>
<td>Social handle, screen name, or alias that identifies this account.</td>
<td>36.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the account as defined by the account’s owner.</td>
<td>36.0</td>
</tr>
<tr>
<td>profileUrl</td>
<td>String</td>
<td>URL to the account’s profile.</td>
<td>36.0</td>
</tr>
<tr>
<td>socialPersonaId</td>
<td>String</td>
<td>ID of the social persona account, if the external social account ID isn’t available.</td>
<td>39.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

ConnectApi.SocialPostCapability
**ConnectApi.SocialAccountRelationship**

Follow relationship between a managed social account and a social persona.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isFollowed</td>
<td>Boolean</td>
<td>Specifies whether the social account is followed by the social persona.</td>
<td>46.0</td>
</tr>
<tr>
<td>isFollowing</td>
<td>Boolean</td>
<td>Specifies whether the social account is following the social persona.</td>
<td>46.0</td>
</tr>
<tr>
<td>socialAccountId</td>
<td>String</td>
<td>ID of the social account.</td>
<td>46.0</td>
</tr>
<tr>
<td>socialPersonaId</td>
<td>String</td>
<td>ID of the social persona.</td>
<td>46.0</td>
</tr>
</tbody>
</table>

**ConnectApi.SocialPostCapability**

If a feed element has this capability, it can interact with a social post on a social network.

Subclass of `ConnectApi.FeedElementCapabilities`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>author</td>
<td><code>ConnectApi.SocialAccount</code></td>
<td>Social account that authored the social post.</td>
<td>36.0</td>
</tr>
<tr>
<td>content</td>
<td>String</td>
<td>Content body of the social post.</td>
<td>36.0</td>
</tr>
<tr>
<td>deletedBy</td>
<td><code>ConnectApi.UserSummary</code></td>
<td>User who deleted the social post.</td>
<td>38.0</td>
</tr>
<tr>
<td>hiddenBy</td>
<td><code>ConnectApi.UserSummary</code></td>
<td>User who hid the social post.</td>
<td>41.0</td>
</tr>
<tr>
<td>icon</td>
<td><code>ConnectApi.Icon</code></td>
<td>Icon of the social network.</td>
<td>36.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID associated with the social post Salesforce record.</td>
<td>36.0</td>
</tr>
<tr>
<td>isOutbound</td>
<td>Boolean</td>
<td>If <code>true</code>, the social post originated from the Salesforce application.</td>
<td>36.0</td>
</tr>
<tr>
<td>likedBy</td>
<td>String</td>
<td>External social account who liked the social post.</td>
<td>40.0</td>
</tr>
<tr>
<td>messageType</td>
<td><code>ConnectApi.SocialPostMessageType</code></td>
<td>Message type of the social post. Values are:</td>
<td>38.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Comment</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Direct</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Post</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• PrivateMessage</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Reply</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Retweet</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Tweet</td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Title or heading of the social post.</td>
<td>36.0</td>
</tr>
<tr>
<td>postUrl</td>
<td>String</td>
<td>External URL to the social post on the social network.</td>
<td>36.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>---------------</td>
<td>------</td>
<td>-------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>provider</td>
<td>ConnectApi.SocialNetworkProvider</td>
<td>Social network that this social post belongs to. Values are: Facebook, GooglePlus, Instagram, InstagramBusiness, KakaoTalk, Kik, Line, LinkedIn, Messenger, Other, Pinterest, QQ, Rypple, SinaWeibo, SMS, Snapchat, Telegram, Twitter, VKontakte, WeChat, WhatsApp, YouTube</td>
<td>36.0</td>
</tr>
<tr>
<td>recipient</td>
<td>ConnectApi.SocialAccount</td>
<td>Social account that is the recipient of the social post.</td>
<td>36.0</td>
</tr>
<tr>
<td>recipientId</td>
<td>String</td>
<td>ID of the recipient of the social post.</td>
<td>38.0</td>
</tr>
<tr>
<td>reviewScale</td>
<td>Double</td>
<td>Review scale of the social post.</td>
<td>40.0</td>
</tr>
<tr>
<td>reviewScore</td>
<td>Double</td>
<td>Review score of the social post.</td>
<td>40.0</td>
</tr>
<tr>
<td>status</td>
<td>ConnectApi.SocialPostStatus</td>
<td>Status of the social post.</td>
<td>36.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.FeedElementCapabilities

**ConnectApi.SocialPostIntents**

Intents available for a social post.
### ConnectApi.SocialPostMassApprovalOutput

Approval or rejection of a large number of social posts.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isApproved</td>
<td>Boolean</td>
<td>Specifies whether the social posts were approved ( \text{true} ) or rejected ( \text{false} ) for publishing.</td>
<td>46.0</td>
</tr>
</tbody>
</table>

### ConnectApi.SocialPostStatus

The status of a social post.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>message</td>
<td>String</td>
<td>Status message.</td>
<td>36.0</td>
</tr>
</tbody>
</table>
### ConnectApi.StatusType

Status type. Values are:
- ApprovalPending
- ApprovalRecalled
- ApprovalRejected
- Deleted
- Failed
- Hidden
- Pending
- Sent
- Unknown

**See also:**
- ConnectApi.SocialPostCapability

### ConnectApi.Stamp

A user stamp.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the stamp.</td>
<td>39.0–43.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the stamp.</td>
<td>39.0–43.0</td>
</tr>
<tr>
<td>imageUrl</td>
<td>String</td>
<td>Image URL of the stamp.</td>
<td>39.0–43.0</td>
</tr>
<tr>
<td>label</td>
<td>String</td>
<td>Label of the stamp.</td>
<td>39.0–43.0</td>
</tr>
</tbody>
</table>

**See also:**
- ConnectApi.User

### ConnectApi.SortRule

Sort rule.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>direction</td>
<td>ConnectApi.CommerceSearch.SortRuleDirectory</td>
<td>Direction of the sort rule. Values are:</td>
<td>52.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Ascending—Sorts in ascending alphanumeric order (A–Z, 0–9).</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Default—When no direction is defined, sorts by relevance.</td>
<td></td>
</tr>
</tbody>
</table>
### Property Name: label
- **Type:** String
- **Description:** Label of the sort rule.
- **Available Version:** 52.0

### Property Name: labelSuffix
- **Type:** `ConnectApi.CommerceSearch.SortRuleLabelSuffix`
- **Description:** Label suffix of the sort rule. Values are:
  - Asc—Label suffix for 'Asc'
  - Ascending—Label suffix for 'Ascending'
  - Az—Label suffix for 'A-Z'
  - Desc—Label suffix for 'Desc'
  - Descending—Label suffix for 'Descending'
  - FewMany—Label suffix for 'Few-Many'
  - HeavyLight—Label suffix for 'Heavy-Light'
  - HighLow—Label suffix for 'High-Low'
  - HighestLowest—Label suffix for 'Highest-Lowest'
  - LightHeavy—Label suffix for 'Light-Heavy'
  - LowHigh—Label suffix for 'Low-High'
  - LowestHighest—Label suffix for 'Lowest-Highest'
  - ManyFew—Label suffix for 'Many-Few'
  - NewOld—Label suffix for 'New-Old'
  - Newest—Label suffix for 'Newest'
  - NewestOldest—Label suffix for 'Newest-Oldest'
  - NineZero—Label suffix for '9-0'
  - OldNew—Label suffix for 'Old-New'
  - Oldest—Label suffix for 'Oldest'
  - OldestNewest—Label suffix for 'Oldest-Newest'
  - PriceDecreasing—Label suffix for '$$-$'
  - PriceIncreasing—Label suffix for '$-$'$
  - ThickThin—Label suffix for 'Thick-Thin'
  - ThinThick—Label suffix for 'Thin-Thick'
  - Za—Label suffix for 'Z-A'
  - ZeroNine—Label suffix for '0-9'
- **Available Version:** 54.0

### Property Name: nameOrId
- **Type:** String
- **Description:** Name of the sort rule field or, if the sort rule is based on a custom field, ID.
- **Available Version:** 52.0
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>sortOrder</td>
<td>Integer</td>
<td>Sort order for the rule. A lower number has higher precedence. The first sort option is called when no other option is selected.</td>
<td>54.0</td>
</tr>
<tr>
<td>sortRuleId</td>
<td>String</td>
<td>ID of the sort rule.</td>
<td>52.0</td>
</tr>
</tbody>
</table>
| type          | ConnectApi.CommerceSearch.SortRuleType | Type of sort rule. Values are:  
- ProductAttributeBased—Sorts by product attribute fields.  
- ProductBased—Sorts by product field data.  
- Relevancy—Sorts by product and catalog term frequency.  
- SortByPricebook—Sorts by product prices defined in the specified pricebook (version 55.0 and later). | 52.0 |

**SEE ALSO:**  
ConnectApi.SortRulesCollection

**ConnectApi.SortRulesCollection**  
Collection of sort rules.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>sortRules</td>
<td>List&lt;ConnectApi.SortRule&gt;</td>
<td>Collection of sort rules.</td>
<td>52.0</td>
</tr>
</tbody>
</table>

**ConnectApi.StatusCapability**  
If a feed post or comment has this capability, it has a status that determines its visibility.  
Subclass of ConnectApi.FeedElementCapability.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
</table>
| feedEntityStatus | ConnectApi.FeedEntityStatus | Status of the feed post or comment. Values are:  
- Draft—The feed post isn’t published but is visible to the author and users with Modify All Data or View All Data permission. Comments can’t be drafts.  
- PendingReview—The feed post or comment isn’t approved yet and therefore isn’t published or visible. | 37.0 |
## Published

- **Published** — The feed post or comment is approved and visible.

### isApprovableByMe

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isApprovableByMe</td>
<td>Boolean</td>
<td>Specifies whether the context user can change the status of the feed post or comment.</td>
<td>37.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- [ConnectApi.CommentCapabilities](#)
- [ConnectApi.FeedElementCapabilities](#)

## ConnectApi.StrategyTrace

Messages and trace nodes for a recommendation strategy execution.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>messages</td>
<td><code>List&lt;String&gt;</code></td>
<td>Messages and errors from the strategy execution.</td>
<td>45.0</td>
</tr>
<tr>
<td>nodes</td>
<td><code>List&lt;ConnectApi.StrategyTraceNode&gt;</code></td>
<td>Nodes of the strategy execution used for debugging.</td>
<td>45.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- [ConnectApi.NBARecommendations](#)

## ConnectApi.StrategyTraceNode

A trace node for a recommendation strategy execution.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>inputCount</td>
<td>Integer</td>
<td>Number of items put into the node.</td>
<td>45.0</td>
</tr>
<tr>
<td>messages</td>
<td><code>List&lt;String&gt;</code></td>
<td>Messages that occurred during node execution.</td>
<td>45.0</td>
</tr>
<tr>
<td>nodeName</td>
<td>String</td>
<td>Name of the node.</td>
<td>45.0</td>
</tr>
<tr>
<td>nodeTime</td>
<td>Long</td>
<td>Time spent processing inside the node.</td>
<td>45.0</td>
</tr>
<tr>
<td>nodeType</td>
<td>String</td>
<td>Type of node.</td>
<td>45.0</td>
</tr>
<tr>
<td>outputCount</td>
<td>Integer</td>
<td>Number of items returned from the node.</td>
<td>45.0</td>
</tr>
<tr>
<td>outputs</td>
<td><code>List&lt;String&gt;</code></td>
<td>Recommendations that are returned from the node.</td>
<td>45.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------------------------</td>
<td>------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>totalTime</td>
<td>Long</td>
<td>Total time spent processing.</td>
<td>45.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.StrategyTrace

### ConnectApi.SubmitCancelOutputRepresentation

ID of the change order created for a cancel action, and a set of its financial values.
Subclass of ConnectApi.BaseOutputRepresentation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>changeBalances</td>
<td>ConnectApi.ChangeItem</td>
<td>Financial values resulting from the cancel.</td>
<td>48.0</td>
</tr>
<tr>
<td>changeOrderId</td>
<td>String</td>
<td>ID of the change order created for the canceled order items and shipping charges. Use this change order to create a credit memo.</td>
<td>48.0</td>
</tr>
<tr>
<td>feeChangeOrderId</td>
<td>String</td>
<td>ID of the change order created by canceling order items with associated cancel fees. Use this change order to create an invoice.</td>
<td>57.0</td>
</tr>
</tbody>
</table>

### ConnectApi.SubmitReturnOutputRepresentation

ID of the change order created for a return action, and a set of its financial values.
Subclass of ConnectApi.BaseOutputRepresentation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>changeBalances</td>
<td>ConnectApi.ChangeItem</td>
<td>Financial values resulting from the return.</td>
<td>48.0</td>
</tr>
<tr>
<td>changeOrderId</td>
<td>String</td>
<td>ID of the change order created for the returned order items and shipping charges. Use this change order to create a credit memo.</td>
<td>48.0</td>
</tr>
<tr>
<td>feeChangeOrderId</td>
<td>String</td>
<td>ID of the change order created by returning order items with associated return fees. Use this change order to create an invoice.</td>
<td>57.0</td>
</tr>
</tbody>
</table>

### ConnectApi.Subscription

Subscription.
### Available Version Description

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>community</td>
<td>ConnectApi.Reference</td>
<td>Information about the Experience Cloud site in which the subscription exists.</td>
<td>28.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>Subscription’s 18–character ID.</td>
<td>28.0</td>
</tr>
<tr>
<td>subject</td>
<td>ConnectApi.Actor</td>
<td>Information about the parent, that is, the thing or person being followed.</td>
<td>28.0</td>
</tr>
<tr>
<td>subscriber</td>
<td>ConnectApi.Actor</td>
<td>Information about the subscriber, that is, the person following this item.</td>
<td>28.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>Connect REST API URL to this specific subscription.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.FollowerPage
- ConnectApi.FollowingPage

### ConnectApi.SubscriptionTermRule

Subscription term rules.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>increment</td>
<td>Integer</td>
<td>Number of pricing term units that can be used to increase the subscription term.</td>
<td>59.0</td>
</tr>
<tr>
<td>maximum</td>
<td>Integer</td>
<td>Maximum number of pricing term units per subscription term.</td>
<td>59.0</td>
</tr>
<tr>
<td>minimum</td>
<td>Integer</td>
<td>Minimum number of pricing term units per subscription term.</td>
<td>59.0</td>
</tr>
</tbody>
</table>

### ConnectApi.SupportedEmojis

A collection of supported emoji.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>supportedEmojis</td>
<td>ConnectApi.EmojiCollection</td>
<td>A collection of supported emoji.</td>
<td>39.0</td>
</tr>
</tbody>
</table>

### ConnectApi.SurveyInvitationEmailOutput

Survey invitation email.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>errorCode</td>
<td>Integer</td>
<td>Error code for the failed call.</td>
<td>50.0</td>
</tr>
</tbody>
</table>
### ConnectApi.OutputClasses

**Available Version**

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>errorMessage</td>
<td>String</td>
<td>Details explaining why the call failed.</td>
<td>50.0</td>
</tr>
</tbody>
</table>
| status        | ConnectApi.SurveyEmailStatusEnum | Status of a survey invitation email. Values are:  
  - Failed—The survey invitation email wasn’t sent.  
  - Queued—The survey invitation email is queued for sending. | 50.0 |

### ConnectApi.Target

Personalization target information.

**Available Version**

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>audience</td>
<td>ConnectApi.AudienceTarget</td>
<td>Audience assigned to the target.</td>
<td>48.0</td>
</tr>
<tr>
<td>formulaScope</td>
<td>ConnectApi.FormulaScope</td>
<td>Formula scope of the target.</td>
<td>50.0</td>
</tr>
<tr>
<td>groupName</td>
<td>String</td>
<td>Group name of the target. Groups bundle related target and audience pairs.</td>
<td>48.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the target.</td>
<td>48.0</td>
</tr>
<tr>
<td>priority</td>
<td>Integer</td>
<td>Priority of the target. Within a group, priority determines which target is returned if the user matches more than one audience.</td>
<td>48.0</td>
</tr>
</tbody>
</table>
| publishStatus | ConnectApi.PublishStatus | Publish status of the target. Values are:  
  - Draft  
  - Live | 48.0 |
| scope         | List<ConnectApi.Scope> | List of scopes for the target.  
In version 50.0 and later, the formulaScope property returns this information. | 48.0–49.0 |
| targetType    | String | Type of target, indicating the nature of the data being targeted. | 48.0 |
| targetValue   | String | Value of the target. | 48.0 |
| url           | String | URL to the target. | 48.0 |

SEE ALSO:

ConnectApi.TargetCollection
**ConnectApi.TargetCollection**

List of personalization targets.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>targets</td>
<td>List&lt;ConnectApi.Target&gt;</td>
<td>List of personalization targets.</td>
<td>48.0</td>
</tr>
</tbody>
</table>

**ConnectApi.CalculateTaxResponse**

Shows the results of a tax calculation request.

Subclass of `ConnectApi.TaxTransactionResponse`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>adapterError</td>
<td>ConnectApi.ErrorResponse</td>
<td>Adapter error.</td>
<td>55.0</td>
</tr>
</tbody>
</table>
| status        | ConnectApi.TaxTransactionStatus | Status of a tax transaction. Values are:  
  - Committed—Tax has been committed to the transaction.  
  - Uncommitted—Tax hasn’t been committed to the transaction. | 55.0 |
| taxEngineLogs | List<ConnectApi.TaxEngineLogResponse> | Tax engine logs. | 55.0 |
| taxTransactionType | ConnectApi.TaxTransactionType | Type of tax transaction. Values are:  
  - Credit—Transaction is a credit transaction.  
  - Debit—Transaction is a debit transaction. | 55.0 |
| taxType       | ConnectApi.CalculateTaxType | Type of tax calculation. Values are:  
  - Actual—Calculated tax represents the final taxed amount for the transaction.  
  - Estimated—Calculated tax represents only an estimated value before the transaction is finalized. | 55.0 |

**ConnectApi.TaxAddressResponse**

Location code of an address.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>locationCode</td>
<td>String</td>
<td>Location code of an address.</td>
<td>55.0</td>
</tr>
</tbody>
</table>
ConnectApi.TaxAddressesResponse

The Ship From, Ship To, and Sold To addresses used during tax calculation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>shipFrom</td>
<td>ConnectApi.TaxAddressResponse</td>
<td>The Ship From address used in tax calculation.</td>
<td>55.0</td>
</tr>
<tr>
<td>shipTo</td>
<td>ConnectApi.TaxAddressResponse</td>
<td>The Ship To address used in tax calculation.</td>
<td>55.0</td>
</tr>
<tr>
<td>soldTo</td>
<td>ConnectApi.TaxAddressResponse</td>
<td>The Sold To address used in tax calculation.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

ConnectApi.TaxAmountDetailsResponse

Information about tax amount values on the line item.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>exemptAmount</td>
<td>Double</td>
<td>Amount of the line item exempt from tax application.</td>
<td>55.0</td>
</tr>
<tr>
<td>taxAmount</td>
<td>Double</td>
<td>Tax amount for the line item.</td>
<td>55.0</td>
</tr>
<tr>
<td>totalAmount</td>
<td>Double</td>
<td>Total amount of the line item.</td>
<td>55.0</td>
</tr>
<tr>
<td>totalAmountWithTax</td>
<td>Double</td>
<td>The line item's total amount with tax.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

ConnectApi.TaxDetailsResponse

Tax details for each line item in a tax line item output.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>exemptAmount</td>
<td>Double</td>
<td>Amount of the line item that is exempt from taxation.</td>
<td>55.0</td>
</tr>
<tr>
<td>exemptReason</td>
<td>String</td>
<td>The reason that any tax exemption applied to the line item.</td>
<td>55.0</td>
</tr>
<tr>
<td>imposition</td>
<td>ConnectApi.TaxImpositionResponse</td>
<td>The business justification for applying tax to a line item.</td>
<td>55.0</td>
</tr>
<tr>
<td>jurisdiction</td>
<td>ConnectApi.TaxJurisdictionResponse</td>
<td>Business address used to calculate the tax rate for the line item.</td>
<td>55.0</td>
</tr>
<tr>
<td>rate</td>
<td>Double</td>
<td>Tax rate for the line item.</td>
<td>55.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>------------------</td>
<td>---------</td>
<td>--------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>tax</td>
<td>Double</td>
<td>Total amount of tax on the line item.</td>
<td>55.0</td>
</tr>
<tr>
<td>taxId</td>
<td>String</td>
<td>ID for the type of tax applied to the line item.</td>
<td>55.0</td>
</tr>
<tr>
<td>taxableAmount</td>
<td>Double</td>
<td>Amount of line item that can be taxed.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

**ConnectApi.TaxAddressResponse**
Address output representation for tax calculation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>locationCode</td>
<td>String</td>
<td>Location code.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

**ConnectApi.TaxEngineLogResponse**
Shows the results of the tax calculation request to the tax engine.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>createdDate</td>
<td>Datetime</td>
<td>The date that the gateway log was created.</td>
<td>55.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the tax engine log record.</td>
<td>55.0</td>
</tr>
<tr>
<td>resultCode</td>
<td>String</td>
<td>Result code sent from the external tax engine. Review the tax engine provider’s documentation for more information about the code.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

**ConnectApi.TaxImpositionResponse**
Tax imposition output representation.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the tax imposition.</td>
<td>55.0</td>
</tr>
<tr>
<td>type</td>
<td>String</td>
<td>Type of the tax imposition.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

**ConnectApi.TaxJurisdictionResponse**
Represents the address or jurisdiction of the primary business used for calculating tax.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>country</td>
<td>String</td>
<td>Country of the tax jurisdiction address.</td>
<td>55.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the tax jurisdiction address.</td>
<td>55.0</td>
</tr>
<tr>
<td>level</td>
<td>String</td>
<td>Level of the tax jurisdiction address.</td>
<td>55.0</td>
</tr>
</tbody>
</table>
ConnectApi.TaxAddressResponse

Available Version
Description
Type
Property Name

55.0
Name of the tax jurisdiction address.
String
name

55.0
Region of the tax jurisdiction address.
String
region

55.0
State-assigned number of the tax jurisdiction address.
String
stateAssignedNo

ConnectApi.LineItemResponse

Response class that stores information about a list of one or more line items on which the tax engine has calculated tax.

Available Version
Description
Type
Property Name

55.0
The Ship From, Ship To, and Sold To addresses used during tax calculation.
ConnectApi.TaxAddressesResponse
addresses

55.0
Information about tax amount values on the line item.
ConnectApi.TaxAmountDetailsResponse
amountDetails

55.0
The date that the tax calculation takes effect.
Datetime
effectiveDate

55.0
System-generated number used to identify the tax line.
String
lineNumber

55.0
Product code for the product related to the taxed line item.
String
productCode

55.0
Quantity of the taxed line item.
Double
quantity

55.0
Tax code for the taxed line item.
String
taxCode

55.0
Tax details for each line item in a tax line item output.
List<ConnectApi.TaxDetailsResponse>
taxes

ConnectApi.TaxTransactionResponse

Tax transaction output representation
This class is abstract.
Superclass of ConnectApi.CalculateTaxResponse.

Available Version
Description
Type
Property Name

55.0
The Ship From, Ship To, and Sold To addresses used during tax calculation.
ConnectApi.TaxAddressesResponse
addresses
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>amountDetails</td>
<td>ConnectApi.TaxAmountDetailsResponse</td>
<td>Information about tax amount values on the line item.</td>
<td>55.0</td>
</tr>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Three-letter ISO 4217 currency code associated with the payment group record.</td>
<td>55.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Information about whether the tax transaction failed or was successful.</td>
<td>55.0</td>
</tr>
<tr>
<td>documentCode</td>
<td>String</td>
<td>Document code.</td>
<td>55.0</td>
</tr>
<tr>
<td>effectiveDate</td>
<td>Datetime</td>
<td>The date that tax is applied to the taxed entity.</td>
<td>55.0</td>
</tr>
<tr>
<td>lineItems</td>
<td>List&lt;ConnectApi.LineItemResponse&gt;</td>
<td>A list of line items on which tax was calculated.</td>
<td>55.0</td>
</tr>
<tr>
<td>referenceDocumentCode</td>
<td>String</td>
<td>The original document code. Used in case of subsequent transactions such as credit tax.</td>
<td>55.0</td>
</tr>
<tr>
<td>referenceEntityId</td>
<td>String</td>
<td>ID of the reference entity used during tax calculation.</td>
<td>55.0</td>
</tr>
<tr>
<td>taxTransactionId</td>
<td>String</td>
<td>ID of the tax transaction.</td>
<td>55.0</td>
</tr>
<tr>
<td>transactionDate</td>
<td>Datetime</td>
<td>The date that the tax transaction occurred.</td>
<td>55.0</td>
</tr>
</tbody>
</table>

**ConnectApi.TextSegment**

Text segment.
Subclass of ConnectApi.MessageSegment.
No additional properties.

**ConnectApi.TimeZone**

User’s time zone as selected in the user’s personal settings in Salesforce. This value doesn’t reflect a device’s current location.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>gmtOffset</td>
<td>Double</td>
<td>Signed offset, in hours, from GMT.</td>
<td>30.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Display name of this time zone.</td>
<td>30.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

ConnectApi.UserSettings

**ConnectApi.Topic**

Topic.
### ConnectApi.ManagedTopic

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>createdDate</td>
<td>Datetime</td>
<td>ISO 8601 date string, for example, 2011-02-25T18:24:31.000Z.</td>
<td>29.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the topic.</td>
<td>29.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>18-character ID.</td>
<td>29.0</td>
</tr>
<tr>
<td>images</td>
<td>ConnectApi.TopicImages</td>
<td>Images associated with the topic.</td>
<td>32.0</td>
</tr>
<tr>
<td>isBeingDeleted</td>
<td>Boolean</td>
<td>true if the topic is currently being deleted; false otherwise.</td>
<td>33.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the topic.</td>
<td>29.0</td>
</tr>
<tr>
<td>nonLocalized</td>
<td>String</td>
<td>Non-localized name of the topic.</td>
<td>36.0</td>
</tr>
<tr>
<td>Name</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>talkingAbout</td>
<td>Integer</td>
<td>Number of people talking about this topic over the last two months, based on factors such as topic additions and comments on posts with the topic.</td>
<td>29.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>URL to the topic detail page.</td>
<td>29.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

- ConnectApi.ManagedTopic
- ConnectApi.TopicPage
- ConnectApi.TopicEndorsement
- ConnectApi.TopicEndorsementCollection
- ConnectApi.TopicSuggestion
- ConnectApi.TopicsCapability

### ConnectApi.TopicEndorsement

Represents one user endorsing another user for a single topic.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>endorsee</td>
<td>ConnectApi.User Summary</td>
<td>User being endorsed.</td>
<td>30.0</td>
</tr>
<tr>
<td>endorsementId</td>
<td>String</td>
<td>18-character ID of the endorsement record.</td>
<td>30.0</td>
</tr>
<tr>
<td>endorser</td>
<td>ConnectApi.User Summary</td>
<td>User performing the endorsement.</td>
<td>30.0</td>
</tr>
<tr>
<td>topic</td>
<td>ConnectApi.Topic</td>
<td>Topic the user is being endorsed for.</td>
<td>30.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>URL to the resource for the endorsement record.</td>
<td>30.0</td>
</tr>
</tbody>
</table>
**ConnectApi.TopicEndorsementCollection**

Collection of topic endorsement response bodies.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>30.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or null if there isn’t a next page.</td>
<td>30.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the previous page, or null if there isn’t a previous page.</td>
<td>30.0</td>
</tr>
<tr>
<td>topicEndorsements</td>
<td>List&lt;ConnectApi.Topic&gt;</td>
<td>List of topic endorsements.</td>
<td>30.0</td>
</tr>
</tbody>
</table>

**ConnectApi.TopicEndorsementSummary**

Topic endorsement summary.

Subclass of ConnectApi.UserActivitySummary.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>endorsementId</td>
<td>ID</td>
<td>ID of the topic endorsement.</td>
<td>42.0</td>
</tr>
</tbody>
</table>

**ConnectApi.TopicImages**

Images associated with a topic.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>coverImageUrl</td>
<td>String</td>
<td>URL to a topic’s cover image, which appears on the topic page. Both topics and managed topics can have cover images.</td>
<td>32.0</td>
</tr>
<tr>
<td>featuredImageUrl</td>
<td>String</td>
<td>URL to a managed topic’s featured image, which appears wherever you feature it, for example, on your Experience Cloud site home page.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

SEE ALSO:

  - ConnectApi.Topic

**ConnectApi.TopicPage**

Page of topics.
### ConnectApi.TopicsCapability

If a feed element has this capability, the context user can add topics to it. Topics help users organize and discover conversations. Subclass of `ConnectApi.FeedElementCapability`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>canAssignTopics</td>
<td>Boolean</td>
<td><code>true</code> if a topic can be assigned to the feed element, <code>false</code> otherwise.</td>
<td>32.0</td>
</tr>
<tr>
<td>items</td>
<td>List&lt;ConnectApi.Topic&gt;</td>
<td>A collection of topics associated with this feed element.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- `ConnectApi.FeedElementCapabilities`

### ConnectApi.TopicSuggestion

Topic suggestion.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>existingTopic</td>
<td>ConnectApi.Topic</td>
<td>Topic that already exists or null for a new topic</td>
<td>29.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Topic name</td>
<td>29.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- `ConnectApi.TopicSuggestionPage`

### ConnectApi.TopicSuggestionPage

Page of topic suggestions.
## List of topic suggestions.

### TopicSuggestions

Summary of a topic.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the topic.</td>
<td>47.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the topic.</td>
<td>47.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

ConnectApi.ManagedContentAssociations

## ConnectApi.TrackedChangeBundleCapability

If a feed element has this capability, it has a group of other feed elements aggregated into one feed element called a **bundle**. This type of bundle aggregates feed tracked changes.

Subclass of ConnectApi.BundleCapability.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>changes</td>
<td>List&lt;String&gt;</td>
<td>Collection of feed tracked changes.</td>
<td>31.0</td>
</tr>
</tbody>
</table>

## ConnectApi.TrackedChangeItem

Tracked change item.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>(fieldName)</td>
<td>String</td>
<td>The name of the field that was updated.</td>
<td>28.0</td>
</tr>
<tr>
<td>newValue</td>
<td>String</td>
<td>The new value of the field or <code>null</code> if the field length is long.</td>
<td>28.0</td>
</tr>
<tr>
<td>oldValue</td>
<td>String</td>
<td>The old value of the field or <code>null</code> if the field length is long.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

ConnectApi.TrackedChangesCapability

ConnectApi.TrackedChangeBundleCapability
**ConnectApi.TrackedChangesCapability**

If a feed element has this capability, it contains all changes to a record for a single tracked change event.
Subclass of **ConnectApi.FeedElementCapability**.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>changes</td>
<td>List&lt;ConnectApi.TrackedChangeItem&gt;</td>
<td>Collection of feed tracked changes.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

SEE ALSO:  
**ConnectApi.FeedElementCapabilities**

**ConnectApi.UnauthenticatedUser**

Unauthenticated user.
Subclass of **ConnectApi.Actor**.
No additional properties.
Instances of this class are used as the actor for feed items and comments posted by Chatter customers.

**ConnectApi.UnreadConversationCount**

Unread count for a conversation.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>hasMore</td>
<td>Boolean</td>
<td>Specifies if there are more than 50 unread messages (true) or not (false).</td>
<td>29.0</td>
</tr>
<tr>
<td>unreadCount</td>
<td>Integer</td>
<td>The total number of unread messages.</td>
<td>29.0</td>
</tr>
</tbody>
</table>

**ConnectApi.UpDownVoteCapability**

If a feed post or comment has this capability, users can upvote or downvote it.
Subclass of **ConnectApi.FeedElementCapability**.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>downVoteCount</td>
<td>Long</td>
<td>Number of downvotes.</td>
<td>41.0</td>
</tr>
<tr>
<td>myVote</td>
<td>ConnectApiUpDownVoteValue</td>
<td>Specifies the context user’s vote. Values are:</td>
<td>41.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Down</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- None</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Up</td>
<td></td>
</tr>
</tbody>
</table>
**ConnectApi.UpVoteSummary**

Summary of an upvote.

Subclass of `ConnectApi.UserFeedEntityActivitySummary`.

No additional properties.

**ConnectApi.User**

User.

This class is abstract.

Subclass of `ConnectApi.ActionWithId`.

Superclass of:

- `ConnectApi.UserDetail`
- `ConnectApi.UserSummary`

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>additionalLabel</td>
<td>String</td>
<td>If one exists, an extra label for the user, for example, &quot;Customer,&quot; &quot;Partner,&quot; or &quot;Acme Corporation.&quot;</td>
<td>30.0</td>
</tr>
<tr>
<td>communityNickname</td>
<td>String</td>
<td>User’s nickname in the site.</td>
<td>32.0</td>
</tr>
<tr>
<td>companyName</td>
<td>String</td>
<td>Name of the company. If your Experience Cloud site allows access without logging in, the value is <code>null</code> for guest users.</td>
<td>28.0</td>
</tr>
<tr>
<td>displayName</td>
<td>String</td>
<td>User’s name that is displayed in the site. If nicknames are enabled, the nickname is displayed. If nicknames aren’t enabled, the full name is displayed.</td>
<td>32.0</td>
</tr>
<tr>
<td>firstName</td>
<td>String</td>
<td>User’s first name. In version 39.0 and later, if nicknames are enabled, <code>firstName</code> is <code>null</code>.</td>
<td>28.0</td>
</tr>
<tr>
<td>isChatterGuest</td>
<td>Boolean</td>
<td><code>true</code> if user is a Chatter customer; <code>false</code> otherwise.</td>
<td>28.0</td>
</tr>
<tr>
<td>isInThisCommunity</td>
<td>Boolean</td>
<td><code>true</code> if user is in the same site as the context user; <code>false</code> otherwise.</td>
<td>28.0</td>
</tr>
<tr>
<td>lastName</td>
<td>String</td>
<td>User’s last name. In version 39.0 and later, if nicknames are enabled, <code>lastName</code> is <code>null</code>.</td>
<td>28.0</td>
</tr>
</tbody>
</table>
### ConnectApi Output Classes

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>outOfOffice</td>
<td>ConnectApi.OutOfOffice</td>
<td>If one exists, extra out-of-office message for the user.</td>
<td>40.0</td>
</tr>
<tr>
<td>photo</td>
<td>ConnectApi.Photo</td>
<td>Information about the user’s photos.</td>
<td>28.0</td>
</tr>
<tr>
<td>reputation</td>
<td>ConnectApi.Reputation</td>
<td>Reputation of the user.</td>
<td>32.0</td>
</tr>
<tr>
<td>stamps</td>
<td>List&lt;ConnectApi.Stamp&gt;</td>
<td>Collection of the user’s stamps.</td>
<td>39.0–43.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>In version 44.0 and later, use SOQL to get a user’s stamps.</td>
<td></td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>User’s title.</td>
<td>28.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If your Experience Cloud site allows access without logging in, the value is null for guest users.</td>
<td></td>
</tr>
<tr>
<td>userType</td>
<td>ConnectApi.UserType Enum</td>
<td>Type of user.</td>
<td>28.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- ChatterGuest—User is an external user in a private group.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- ChatterOnly—User is a Chatter Free customer.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Guest—User is unauthenticated.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Internal—User is a standard org member.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Portal—User is an external user in an Experience Cloud site.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- System—User is Chatter Expert or a system user.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Undefined—User is a user type that is a custom object.</td>
<td></td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.RecommendationAudience

### ConnectApi.UserActivitiesJob

User activities job.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>jobToken</td>
<td>String</td>
<td>Token that identifies the user activities job.</td>
<td>42.0</td>
</tr>
<tr>
<td>jobType</td>
<td>String</td>
<td>Type of user activities job. Value is export or purge.</td>
<td>42.0</td>
</tr>
</tbody>
</table>
### ConnectApi.UserActivityCollection

User activity collection.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>activityType</td>
<td>String</td>
<td>Type of user activity. Values are:</td>
<td>42.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Bookmark—User bookmarked a post.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ChatterActivity—Total counts of posts and comments made and likes and comments received for a user.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• ChatterLike—User liked a post or comment.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Comment—User commented on a post.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• CompanyVerify—User verified comment.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• DownVote—User downvoted a post or comment.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FeedEntityRead—User read a post.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• FeedRead—User read a feed.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Mute—User muted a post.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Post—User made a post.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• TopicEndorsement—User endorsed another user on a topic or received endorsement on a topic.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• UpVote—User upvoted a post or comment.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>userActivities</td>
<td>List&lt;ConnectApi.UserActivitySummary&gt;</td>
<td>Collection of user activities.</td>
<td>42.0</td>
</tr>
</tbody>
</table>

### ConnectApi.UserActivitySummary

User activity summary.

This class is abstract.

Superclass of:

- ConnectApi.CommentSummary
### ConnectApi.OutputClasses

**ConnectApi.FeedPostSummary**

**ConnectApi.FeedReadSummary**

**ConnectApi.TopicEndorsementSummary**

**ConnectApi.UserFeedEntityActivitySummary**

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>activityDate</td>
<td>Datetime</td>
<td>Date of the user activity.</td>
<td>42.0</td>
</tr>
<tr>
<td>activityType</td>
<td>String</td>
<td>Type of user activity. Values are:</td>
<td>42.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Bookmark—User bookmarked a post.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- ChatterActivity—Total counts of posts and comments made and likes and comments received for a user.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- ChatterLike—User liked a post or comment.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Comment—User commented on a post.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- CompanyVerify—User verified comment.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- DownVote—User downvoted a post or comment.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- FeedEntityRead—User read a post.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- FeedRead—User read a feed.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Mute—User muted a post.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Post—User made a post.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- TopicEndorsement—User endorsed another user on a topic or received endorsement on a topic.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- UpVote—User upvoted a post or comment.</td>
<td></td>
</tr>
<tr>
<td>activityUrl</td>
<td>String</td>
<td>URL to the user activity.</td>
<td>42.0</td>
</tr>
<tr>
<td>community</td>
<td>ConnectApi.</td>
<td>Experience Cloud site in which the user performed the activity.</td>
<td>42.0</td>
</tr>
<tr>
<td></td>
<td>CommunitySummary</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

SEE ALSO:

- **ConnectApi.UserActivityCollection**

### ConnectApi.UserCapabilities

Capabilities associated with a user profile.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>canChat</td>
<td>Boolean</td>
<td>Specifies if the context user can use Chatter Messenger with the subject user (true) or not (false)</td>
<td>29.0</td>
</tr>
</tbody>
</table>
canDirectMessage: Boolean
   Specifies if the context user can direct message the subject user (true) or not (false)
   Available Version: 29.0

canEdit: Boolean
   Specifies if the context user can edit the subject user's account (true) or not (false)
   Available Version: 29.0

canFollow: Boolean
   Specifies if the context user can follow the subject user's feed (true) or not (false)
   Available Version: 29.0

canViewFeed: Boolean
   Specifies if the context user can view the feed of the subject user (true) or not (false)
   Available Version: 29.0

canViewFullProfile: Boolean
   Specifies if the context user can view the full profile of the subject user (true) or only the limited profile (false)
   Available Version: 29.0

isModerator: Boolean
   Specifies if the subject user is a Chatter moderator or admin (true) or not (false)
   Available Version: 29.0

SEE ALSO:
   ConnectApi.UserProfile

ConnectApi.UserChatterSettings
User's global Chatter settings.

defaultGroup
   ConnectApi.GroupEmail EmailFrequency
   The default frequency with which a user receives email from a group when they join it.
   Available Version: 28.0

ConnectApi.UserDetail
Details about a user in an org.
Subclass of ConnectApi.User.
If the context user doesn't have permission to see a property, its value is set to null.

aboutMe: String
   Text from user's profile.
   Available Version: 28.0

address: ConnectApi.Address
   User's address.
   Available Version: 28.0

bannerPhoto: ConnectApi.BannerPhoto
   User's banner photo.
   Available Version: 36.0

chatterActivity: ConnectApi.ChatterActivity
   Chatter activity statistics.
   Available Version: 28.0
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>chatterInfluence</td>
<td><code>ConnectApi.GlobalInfluence</code></td>
<td>User’s influence rank.</td>
<td>28.0</td>
</tr>
<tr>
<td>email</td>
<td><code>String</code></td>
<td>User’s email address.</td>
<td>28.0</td>
</tr>
<tr>
<td>followersCount</td>
<td><code>Integer</code></td>
<td>Number of users following this user.</td>
<td>28.0</td>
</tr>
<tr>
<td>followingCounts</td>
<td><code>ConnectApi.FollowingCounts</code></td>
<td>Information about items the user is following.</td>
<td>28.0</td>
</tr>
<tr>
<td>groupCount</td>
<td><code>Integer</code></td>
<td>Number of groups user is following.</td>
<td>28.0</td>
</tr>
<tr>
<td>hasChatter</td>
<td><code>Boolean</code></td>
<td><code>true</code> if user has access to Chatter; <code>false</code> otherwise.</td>
<td>31.0</td>
</tr>
<tr>
<td>isActive</td>
<td><code>Boolean</code></td>
<td><code>true</code> if user is active; <code>false</code> otherwise.</td>
<td>28.0</td>
</tr>
<tr>
<td>managerId</td>
<td><code>String</code></td>
<td>18-character ID of the user’s manager.</td>
<td>28.0</td>
</tr>
<tr>
<td>managerName</td>
<td><code>String</code></td>
<td>Locale-based concatenation of manager’s first and last names.</td>
<td>28.0</td>
</tr>
<tr>
<td>phoneNumbers</td>
<td><code>List&lt;ConnectApi.PhoneNumber&gt;</code></td>
<td>Collection of user’s phone numbers.</td>
<td>28.0</td>
</tr>
<tr>
<td>thanksReceived</td>
<td><code>Integer</code></td>
<td>The number of times the user has been thanked.</td>
<td>29.0</td>
</tr>
<tr>
<td>username</td>
<td><code>String</code></td>
<td>Username of the user, such as <code>Admin@mycompany.com</code>.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- `ConnectApi.UserPage`
- `ConnectApi.UserProfile`

**ConnectApi.UserFeedEntityActivitySummary**

User feed entity activity summary.

This class is abstract.

Subclass of `ConnectApi.UserActivitySummary`.

Superclass of:
- `ConnectApi.BookmarkSummary`
- `ConnectApi.ChatterActivitySummary`
- `ConnectApi.CompanyVerifySummary`
- `ConnectApi.DownVoteSummary`
- `ConnectApi.FeedEntityReadSummary`
- `ConnectApi.LikeSummary`
- `ConnectApi.MuteSummary`
ConnectApi.UpVoteSummary

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>feedEntityId</td>
<td>String</td>
<td>ID of the feed entity.</td>
<td>42.0</td>
</tr>
</tbody>
</table>

ConnectApi.UserGroupDetailPage

A page of groups that a user is a member of.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>URL to the current page.</td>
<td>45.0</td>
</tr>
<tr>
<td>groups</td>
<td>List&lt;ConnectApi.ChatterGroupDetail&gt;</td>
<td>Collection of groups that the user is a member of.</td>
<td>45.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>URL to the next page, or null if there is no next page.</td>
<td>45.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>URL to the previous page, or null if there is no previous page.</td>
<td>45.0</td>
</tr>
<tr>
<td>total</td>
<td>Integer</td>
<td>Total number of groups that the user is a member of.</td>
<td>45.0</td>
</tr>
</tbody>
</table>

ConnectApi.UserGroupPage

A paginated list of groups the context user is a member of.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>28.0</td>
</tr>
<tr>
<td>groups</td>
<td>List&lt;ConnectApi.ChatterGroupSummary&gt;</td>
<td>List of groups.</td>
<td>28.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or null if there isn’t a next page.</td>
<td>28.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the previous page, or null if there isn’t a previous page.</td>
<td>28.0</td>
</tr>
<tr>
<td>total</td>
<td>Integer</td>
<td>Total number of groups across all pages.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

ConnectApi.UserMission

Mission details for a user.

### ConnectApi.UserMissionActivitiesJob

User mission activities job.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>jobToken</td>
<td>String</td>
<td>Token that identifies the mission user activities job.</td>
<td>45.0</td>
</tr>
<tr>
<td>jobType</td>
<td>String</td>
<td>Type of user activities job, either <code>export</code> or <code>purge</code>.</td>
<td>45.0</td>
</tr>
<tr>
<td>message</td>
<td>String</td>
<td>Message describing the status and expected outcome of the job.</td>
<td>45.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>When the job completes, you receive an email with information about the Salesforce file that contains ConnectApi.UserMissionActivityCollection.</td>
<td></td>
</tr>
</tbody>
</table>

### ConnectApi.UserMissionActivity

User activity associated with missions.


No additional properties.

### ConnectApi.UserMissionActivityCollection

List of mission activities for a user.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>community</td>
<td>ConnectApi.CommunitySummary</td>
<td>Experience Cloud site in which the user performed activities.</td>
<td>45.0</td>
</tr>
<tr>
<td>userId</td>
<td>String</td>
<td>ID of the user.</td>
<td>45.0</td>
</tr>
<tr>
<td>userMissionActivities</td>
<td>List&lt;ConnectApi.AbstractUserMissionActivity&gt;</td>
<td>List of mission activities performed by the user.</td>
<td>45.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>----------------</td>
<td>---------</td>
<td>--------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>userName</td>
<td>String</td>
<td>Name of the user.</td>
<td>45.0</td>
</tr>
</tbody>
</table>

SEE ALSO:  
ConnectApi.UserMissionActivitiesJob

**ConnectApi.UserMissionActivityStatus**

Status of mission activity for a user.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>message</td>
<td>String</td>
<td>Success or error message.</td>
<td>45.0</td>
</tr>
<tr>
<td>status</td>
<td>String</td>
<td>Status of mission activity for a user.</td>
<td>45.0</td>
</tr>
</tbody>
</table>

**ConnectApi.UserOauthInfo**

User OAuth information.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>availableExternal</td>
<td>Connect.Oauth</td>
<td>ProviderInfo</td>
<td>37.0</td>
</tr>
<tr>
<td>EmailService</td>
<td>ProviderInfo</td>
<td></td>
<td></td>
</tr>
<tr>
<td>isAuthenticated</td>
<td>Boolean</td>
<td>Specifies whether the user is authenticated (true) or not (false).</td>
<td>37.0</td>
</tr>
</tbody>
</table>

**ConnectApi.UserPage**

Page of users.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageToken</td>
<td>Integer</td>
<td>Token identifying the current page.</td>
<td>28.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>28.0</td>
</tr>
<tr>
<td>nextPageToken</td>
<td>Integer</td>
<td>Token identifying the next page, or null if there isn’t a next page.</td>
<td>28.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or null if there isn’t a next page.</td>
<td>28.0</td>
</tr>
<tr>
<td>previousPageToken</td>
<td>Integer</td>
<td>Token identifying the previous page, or null if there isn’t a previous page.</td>
<td>28.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the previous page, or null if there isn’t a previous page.</td>
<td>28.0</td>
</tr>
</tbody>
</table>
## ConnectApi.UserProfile
Details necessary to render a view of a user profile.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>capabilities</td>
<td>ConnectApi.UserCapabilities</td>
<td>The context user’s capabilities specific to the subject user’s profile.</td>
<td>29.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>The ID of the user attached to the profile.</td>
<td>29.0</td>
</tr>
<tr>
<td>tabs</td>
<td>List&lt;ConnectApi.UserProfileTab&gt;</td>
<td>The tabs visible to the context user specific to the subject user’s profile.</td>
<td>29.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>The URL of the user’s profile.</td>
<td>29.0</td>
</tr>
<tr>
<td>userDetail</td>
<td>ConnectApi.UserDetail</td>
<td>The details about the user attached to the profile.</td>
<td>29.0</td>
</tr>
</tbody>
</table>

## ConnectApi.UserProfileTab
Information about a profile tab.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>The tab’s unique identifier or 18–character ID</td>
<td>29.0</td>
</tr>
<tr>
<td>isDefault</td>
<td>Boolean</td>
<td>Specifies if the tab appears first when clicking the user profile (true) or not (false)</td>
<td>29.0</td>
</tr>
<tr>
<td>tabType</td>
<td>ConnectApi.UserProfile TabType Enum</td>
<td>Specifies the type of tab</td>
<td>29.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• CustomVisualForce—Tab that displays data from a Visualforce page.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• CustomWeb—Tab that displays data from any external web-based application or web page.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Element—Tab that displays generic content inline.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Feed—Tab that displays the Chatter feed.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Overview—Tab that displays user details.</td>
<td></td>
</tr>
</tbody>
</table>
### ConnectApi.UserProfile

A list of user references.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>URL to the current page.</td>
<td>35.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>URL to the next page.</td>
<td>35.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>URL to the previous page.</td>
<td>35.0</td>
</tr>
<tr>
<td>userCount</td>
<td>Integer</td>
<td>Number of users in the collection.</td>
<td>35.0</td>
</tr>
<tr>
<td>users</td>
<td>List&lt;ConnectApi.Reference&gt;</td>
<td>A collection of user references.</td>
<td>35.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.CustomListAudienceCriteria

### ConnectApi.UserReferencePage

A list of user references.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>tabUrl</td>
<td>String</td>
<td>The current tab’s content URL (for non built-in tab types)</td>
<td>29.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.UserProfile

### ConnectApi.UserSettings

Settings specific to a user.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>approvalPosts</td>
<td>Boolean</td>
<td>User can approve workflows from Chatter posts.</td>
<td>28.0</td>
</tr>
<tr>
<td>canAccessPersonalStreams</td>
<td>Boolean</td>
<td>User can access personal stream feeds.</td>
<td>40.0</td>
</tr>
<tr>
<td>canFollow</td>
<td>Boolean</td>
<td>User can follow users and records.</td>
<td>28.0</td>
</tr>
<tr>
<td>canModifyAllData</td>
<td>Boolean</td>
<td>User has Modify all Data permission.</td>
<td>28.0</td>
</tr>
<tr>
<td>canOwnGroups</td>
<td>Boolean</td>
<td>User can own groups.</td>
<td>28.0</td>
</tr>
<tr>
<td>canViewAllData</td>
<td>Boolean</td>
<td>User has View all Data permission.</td>
<td>28.0</td>
</tr>
<tr>
<td>canViewAllGroups</td>
<td>Boolean</td>
<td>User has View all Groups permission.</td>
<td>28.0</td>
</tr>
<tr>
<td>Property</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>--------------------------</td>
<td>---------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>canViewAllUsers</td>
<td>Boolean</td>
<td>User has View all Users permission.</td>
<td>28.0</td>
</tr>
<tr>
<td>canViewCommunitySwitcher</td>
<td>Boolean</td>
<td>User can see the site switcher menu.</td>
<td>34.0</td>
</tr>
<tr>
<td>canViewFullUserProfile</td>
<td>Boolean</td>
<td>User can see other user's Chatter profile.</td>
<td>28.0</td>
</tr>
<tr>
<td>canViewPublicFiles</td>
<td>Boolean</td>
<td>User can see all files that are public.</td>
<td>28.0</td>
</tr>
<tr>
<td>currencySymbol</td>
<td>String</td>
<td>Currency symbol to use for displaying currency values. Applicable only when the ConnectApi.Features.multiCurrency property is false.</td>
<td>28.0</td>
</tr>
<tr>
<td>externalUser</td>
<td>Boolean</td>
<td>User is a Chatter customer.</td>
<td>28.0</td>
</tr>
<tr>
<td>fileSyncLimit</td>
<td>Integer</td>
<td>Maximum number of files user can sync.</td>
<td>32.0</td>
</tr>
<tr>
<td>fileSyncStorageLimit</td>
<td>Integer</td>
<td>Maximum storage for synced files, in megabytes (MB).</td>
<td>29.0</td>
</tr>
<tr>
<td>folderSyncLimit</td>
<td>Integer</td>
<td>Maximum number of folders user can sync.</td>
<td>32.0</td>
</tr>
<tr>
<td>hasAccessToInternalOrg</td>
<td>Boolean</td>
<td>User is a member of the internal org.</td>
<td>28.0</td>
</tr>
<tr>
<td>hasChatter</td>
<td>Boolean</td>
<td>User has access to Chatter.</td>
<td>31.0</td>
</tr>
<tr>
<td>hasFileSync</td>
<td>Boolean</td>
<td>User has Sync Files permission.</td>
<td>28.0</td>
</tr>
<tr>
<td>hasFieldServiceLocationTracking</td>
<td>Boolean</td>
<td>User has Field Service GPS tracking enabled.</td>
<td>41.0</td>
</tr>
<tr>
<td>hasFieldServiceMobileAccess</td>
<td>Boolean</td>
<td>User has access to the Field Service mobile app.</td>
<td>41.0</td>
</tr>
<tr>
<td>hasFileSyncManagedClientAutoUpdate</td>
<td>Boolean</td>
<td>Administrator for the user's org allows file sync clients to update automatically.</td>
<td>34.0</td>
</tr>
<tr>
<td>hasRestDataApiAccess</td>
<td>Boolean</td>
<td>User has access to REST API.</td>
<td>29.0</td>
</tr>
<tr>
<td>timeZone</td>
<td>ConnectApi.TimeZone</td>
<td>The user's time zone as selected in the user's personal settings in Salesforce. This value does not reflect a device's current location.</td>
<td>30.0</td>
</tr>
<tr>
<td>userDefaultCurrencyIsoCode</td>
<td>String</td>
<td>The ISO code for the default currency. Applicable only when the ConnectApi.Features.multiCurrency property is true.</td>
<td>28.0</td>
</tr>
<tr>
<td>userId</td>
<td>String</td>
<td>18-character ID of the user.</td>
<td>28.0</td>
</tr>
<tr>
<td>Property</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>-----------</td>
<td>---------</td>
<td>------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>userLocale</td>
<td>String</td>
<td>Locale of user.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.OrganizationSettings

**ConnectApi.UserSummary**

User summary.
Subclass of ConnectApi.User.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isActive</td>
<td>Boolean</td>
<td>true if user is active; false otherwise.</td>
<td>28.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.ChatterConversation
- ConnectApi.ChatterConversationSummary
- ConnectApi.ChatterGroup
- ConnectApi.ChatterLike
- ConnectApi.DashboardComponentSnapshot
- ConnectApi.DirectMessageMemberPage
- ConnectApi.GroupMembershipRequest
- ConnectApi.GroupMember
- ConnectApi.FeedFavorite
- ConnectApi.OriginCapability
- ConnectApi.PlatformAction
- ConnectApi.DirectMessageMemberPage
- ConnectApi.DirectMessageMemberActivity
- ConnectApi.ChatterMessage
- ConnectApi.Comment
- ConnectApi.File
- ConnectApi.MentionSegment
- ConnectApi.QuestionAndAnswersCapability
- ConnectApi.SocialPostCapability
- ConnectApi.TopicEndorsement
**ConnectApi.VerifiedCapability**

If a comment has this capability, users with permission can mark it as verified or unverified.

Subclass of **ConnectApi.FeedElementCapability**.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>isVerifiableByMe</td>
<td>Boolean</td>
<td>Specifies whether the context user has permission to mark comments as verified or unverified</td>
<td>41.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(true) or not (false).</td>
<td></td>
</tr>
<tr>
<td>isVerified</td>
<td>Boolean</td>
<td>Specifies whether the comment is marked as verified (true) or not (false).</td>
<td>41.0</td>
</tr>
<tr>
<td>isVerifiedByAnonymized</td>
<td>Boolean</td>
<td>Specifies whether the comment is marked as verified by an anonymous user (true) or not (false).</td>
<td>43.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If the comment has never been marked as verified or unverified, null. Also null if the context</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>user doesn’t have permission to mark comments as verified or unverified.</td>
<td></td>
</tr>
<tr>
<td>lastVerifiedByUser</td>
<td>ConnectApi.UserSummary</td>
<td>User who last marked the comment as verified or unverified, otherwise null. Also null if the context user doesn’t have permission to mark comments as verified or unverified.</td>
<td>41.0</td>
</tr>
<tr>
<td>lastVerifiedDate</td>
<td>Datetime</td>
<td>Date when the comment was last marked as verified or unverified, otherwise null. Also null if</td>
<td>41.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>the context user doesn’t have permission to mark comments as verified or unverified.</td>
<td></td>
</tr>
</tbody>
</table>

**SEE ALSO:**

**ConnectApi.CommentCapabilities**

**ConnectApi.Vote**

An upvote or downvote on a feed element or comment.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>ConnectApiUpDownVoteValue</td>
<td>Type of vote for a feed element or comment.</td>
<td>42.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Down</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Up</td>
<td></td>
</tr>
<tr>
<td>user</td>
<td>ConnectApiUserSummary</td>
<td>User who voted on the feed element or comment.</td>
<td>42.0</td>
</tr>
</tbody>
</table>
### ConnectApi.VotePage

A page of upvotes or downvotes on a feed element or comment.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageToken</td>
<td>Integer</td>
<td>Token identifying the current page.</td>
<td>42.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>42.0</td>
</tr>
<tr>
<td>items</td>
<td>List&lt;ConnectApi.Vote&gt;</td>
<td>Collection of users and their upvotes or downvotes. Upvotes include likes and upvotes. For example, if a post receives five likes and three upvotes, the number of upvotes is eight. For this reason, the collection of users and their upvotes also includes users who liked the post or comment. If a user both liked and upvoted a post, they appear only once in the collection.</td>
<td>42.0</td>
</tr>
<tr>
<td>nextPageToken</td>
<td>Integer</td>
<td>Token identifying the next page, or null if there isn’t a next page.</td>
<td>42.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or null if there isn’t a next page.</td>
<td>42.0</td>
</tr>
<tr>
<td>previousPageToken</td>
<td>Integer</td>
<td>Token identifying the previous page, or null if there isn’t a previous page.</td>
<td>42.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the previous page, or null if there isn’t a previous page.</td>
<td>42.0</td>
</tr>
<tr>
<td>total</td>
<td>Long</td>
<td>Total number of upvotes or downvotes for the feed element or comment. The number of upvotes includes the number of likes and upvotes. For example, if a post receives five likes and three upvotes, the total number of upvotes is eight. If a user both liked and upvoted a post, we count that as two upvotes.</td>
<td>42.0</td>
</tr>
</tbody>
</table>

### ConnectApi.Wishlist

Wishlist, including summary and items.
### Available Version

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>page</td>
<td>ConnectApi.WishlistItemCollection</td>
<td>Page of wishlist items.</td>
<td>49.0</td>
</tr>
<tr>
<td>summary</td>
<td>ConnectApi.WishlistSummary</td>
<td>Summary of the wishlist.</td>
<td>49.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.WishlistsSummary

### ConnectApi.WishlistItem

Item in a wishlist.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Three-letter ISO 4217 currency code associated with the product.</td>
<td>49.0</td>
</tr>
<tr>
<td>error</td>
<td>ConnectApi.ErrorResponse</td>
<td>Error information.</td>
<td>49.0</td>
</tr>
<tr>
<td>listPrice</td>
<td>Double</td>
<td>List price of the wishlist item.</td>
<td>49.0</td>
</tr>
<tr>
<td>productSummary</td>
<td>ConnectApiCartItemProduct</td>
<td>Product summary for the wishlist item.</td>
<td>49.0</td>
</tr>
<tr>
<td>salesPrice</td>
<td>Double</td>
<td>Sales price of the wishlist item.</td>
<td>49.0</td>
</tr>
<tr>
<td>wishlistItemId</td>
<td>String</td>
<td>ID of the wishlist item.</td>
<td>49.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
- ConnectApi.WishlistItemCollection

### ConnectApi.WishlistItemCollection

Collection of wishlist items.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currencyIsoCode</td>
<td>String</td>
<td>Three-letter ISO 4217 currency code associated with the product.</td>
<td>49.0</td>
</tr>
<tr>
<td>currentPageToken</td>
<td>String</td>
<td>Token identifying the current page.</td>
<td>49.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>49.0</td>
</tr>
<tr>
<td>hasErrors</td>
<td>Boolean</td>
<td>Specifies whether at least one of the results contains an error (true) or not (false).</td>
<td>49.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>-----------------</td>
<td>-------------------------------</td>
<td>-------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>items</td>
<td>List&lt;ConnectApi.WishlistItem&gt;</td>
<td>Collection of wishlist items.</td>
<td>49.0</td>
</tr>
<tr>
<td>nextPageToken</td>
<td>String</td>
<td>Token identifying the next page, or null if there isn’t a next page.</td>
<td>49.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or null if there isn’t a next page.</td>
<td>49.0</td>
</tr>
<tr>
<td>previousPageToken</td>
<td>String</td>
<td>Token identifying the previous page, or null if there isn’t a previous page.</td>
<td>49.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the previous page, or null if there isn’t a previous page.</td>
<td>49.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**
- ConnectApi.Wishlist

**ConnectApi.WishlistsSummary**
List of wishlist summaries and the displayed list for the context user.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>displayedList</td>
<td>ConnectApi.Wishlist</td>
<td>Oldest wishlist displayed for the context user.</td>
<td>49.0</td>
</tr>
<tr>
<td>summaries</td>
<td>List&lt;ConnectApi.WishlistSummary&gt;</td>
<td>Summary of wishlists belonging to the context user.</td>
<td>49.0</td>
</tr>
<tr>
<td>wishlistCount</td>
<td>Integer</td>
<td>Total number of wishlists belonging to the context user.</td>
<td>49.0</td>
</tr>
</tbody>
</table>

**ConnectApi.WishlistSummary**
Summary of a wishlist.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>createdDate</td>
<td>Datetime</td>
<td>Created date for the wishlist in ISO 8601 format, for example, 2011-02-25T18:24:31.000Z.</td>
<td>49.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the wishlist.</td>
<td>49.0</td>
</tr>
<tr>
<td>modifiedDate</td>
<td>Datetime</td>
<td>Last modified date of the wishlist in ISO 8601 format, for example, 2011-02-25T18:24:31.000Z.</td>
<td>49.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the wishlist.</td>
<td>49.0</td>
</tr>
</tbody>
</table>
### ConnectApi.WishlistToCartResult

Result of adding a wishlist to a cart.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>cartId</td>
<td>String</td>
<td>ID of the cart to which the products were added.</td>
<td>49.0</td>
</tr>
<tr>
<td>failedWishlistToCartItems</td>
<td>List&lt;ConnectApiCartItemResult&gt;</td>
<td>Wishlist items that weren’t successfully added to the cart.</td>
<td>49.0</td>
</tr>
<tr>
<td>productsFailedCount</td>
<td>Integer</td>
<td>Total number of products that weren’t added to the cart.</td>
<td>49.0</td>
</tr>
<tr>
<td>productsRequestedCount</td>
<td>Integer</td>
<td>Total number of products requested to add to the cart.</td>
<td>49.0</td>
</tr>
<tr>
<td>productsSucceededCount</td>
<td>Integer</td>
<td>Total number of products that were successfully added to the cart.</td>
<td>49.0</td>
</tr>
<tr>
<td>succeededWishlistToCartItems</td>
<td>List&lt;ConnectApiCartItemResult&gt;</td>
<td>Wishlist items that were successfully added to the cart.</td>
<td>49.0</td>
</tr>
</tbody>
</table>

### ConnectApi.Zone

Information about a Chatter Answers zone.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>String</td>
<td>The description of the zone.</td>
<td>29.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>The zone ID.</td>
<td>29.0</td>
</tr>
<tr>
<td>isActive</td>
<td>Boolean</td>
<td>Indicates whether or not the zone is active.</td>
<td>29.0</td>
</tr>
<tr>
<td>isChatterAnswers</td>
<td>Boolean</td>
<td>Indicates whether or not the zone is available for Chatter Answers.</td>
<td>29.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>Name of the zone.</td>
<td>29.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>The URL of the zone.</td>
<td>30.0</td>
</tr>
</tbody>
</table>
## Available Version Description

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>visibility</td>
<td>ConnectApi.ZoneShowIn</td>
<td>Zone visibility type.</td>
<td>29.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Community—Available in an Experience Cloud site.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Internal—Available internally only.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Portal—Available in a portal.</td>
<td></td>
</tr>
<tr>
<td>visibilityId</td>
<td>String</td>
<td>If the zone is available in a site, this property contains the ID of the site. If the zone is available to all sites, this property contains the value All.</td>
<td>29.0</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

ConnectApi.ZonePage

### ConnectApi.ZonePage

Page of zones.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>zones</td>
<td>List&lt;ConnectApi.Zone&gt;</td>
<td>A list of one or more zones.</td>
<td>29.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>29.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or null if there isn’t a next page.</td>
<td>29.0</td>
</tr>
</tbody>
</table>

### ConnectApi.ZoneSearchPage

Page of zone search results.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageToken</td>
<td>String</td>
<td>Token identifying the current page.</td>
<td>29.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>29.0</td>
</tr>
<tr>
<td>items</td>
<td>List&lt;ConnectApi.ZoneSearchResult&gt;</td>
<td>List of search results.</td>
<td>29.0</td>
</tr>
<tr>
<td>nextPageToken</td>
<td>String</td>
<td>Token identifying the next page, or null if there isn’t a next page.</td>
<td>29.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or null if there isn’t a next page.</td>
<td>29.0</td>
</tr>
</tbody>
</table>
ConnectApi.ZoneSearchResult

Information about a specific zone search result.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>hasBestAnswer</td>
<td>Boolean</td>
<td>Indicates if the search result has a best answer.</td>
<td>29.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>ID of the search result. The search result can be a question or an article.</td>
<td>29.0</td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>Title of the search result.</td>
<td>29.0</td>
</tr>
<tr>
<td>type</td>
<td>ConnectApi.ZoneSearchResultTypeEnum</td>
<td>Specifies the zone search result type.</td>
<td>29.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Article—Search results contain only articles.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Question—Search results contain only questions.</td>
<td></td>
</tr>
<tr>
<td>voteCount</td>
<td>String</td>
<td>Number of votes given to the search result.</td>
<td>29.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.ZoneSearchPage

Retired ConnectApi Output Classes

These ConnectApi output classes are retired.

IN THIS SECTION:
- ConnectApi.ApprovalAttachment
  Attach an approval to a feed item.
- ConnectApi.BasicTemplateAttachment
  Attachments in feed items with type BasicTemplate.
- ConnectApi.CanvasTemplateAttachment
  Attachments in feed items with type CanvasPost.
- ConnectApi.CaseComment
  Attachments in feed items with type CaseCommentPost.
- ConnectApi.ContentAttachment
  Attachments in feed items with type ContentPost.
- ConnectApi.DashboardComponentAttachment
  Attachments in feed items with type DashboardSnapshot.
- ConnectApi.DatacloudCompany
  Information about a Data.com company.
ConnectApi.DatacloudCompanies
Lists all companies that were purchased in a specific order, page URLs, and the number of companies in the order.

ConnectApi.DatacloudContact
Information about a Data.com contact.

ConnectApi.DatacloudContacts
Lists all contacts that were purchased in the specific order, page URLs, and the number of contacts in the order.

ConnectApi.DatacloudOrder
Represents a Datacloud order.

ConnectApi.DatacloudPurchaseUsage
Information about Data.com point usage for monthly and list pool users.

ConnectApi.EmailMessage
Email message from a case.

ConnectApi.FeedItemAttachment
Feed item attachment.

ConnectApi.FeedItemPage
A paged collection of ConnectApi.FeedItem objects.

ConnectApi.FeedItemTopicPage
Feed item topic page.

ConnectApi.FeedPoll
Attachment of ConnectApi.FeedItem objects where the type property is PollPost.

ConnectApi.LinkAttachment
Link attached to a feed item.

ConnectApi.NonEntityRecommendation
A recommendation for a non-Salesforce entity, such as an application.

ConnectApi.RecordSnapshotAttachment
Fields of a record at the point in time when the record was created.

ConnectApi.TrackedChangeAttachment
Tracked change attachment to a feed item.

ConnectApi.ApprovalAttachment
Attach an approval to a feed item.

⚠️ Important: This class isn’t available in version 32.0 and later. In version 32.0 and later, ConnectApi.ApprovalCapability is used.

Subclass of ConnectApi.FeedItemAttachment.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>String</td>
<td>A work item ID.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>postTemplateFields</td>
<td>List (&lt;ConnectApi.ApprovalPostTemplateField&gt;)</td>
<td>Collection of approval post template fields</td>
<td>28.0–31.0</td>
</tr>
</tbody>
</table>
### Available Version

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>process InstanceStepId</td>
<td>String</td>
<td>An approval step ID.</td>
<td>30.0–31.0</td>
</tr>
<tr>
<td>status</td>
<td>ConnectApi.WorkflowProcess Status Enum</td>
<td>Status of a workflow process.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Approved</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Fault</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Held</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• NoResponse</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Pending</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Reassigned</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Rejected</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Removed</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Started</td>
<td></td>
</tr>
</tbody>
</table>

### ConnectApi.BasicTemplateAttachment

Attachments in feed items with type BasicTemplate.

⚠️ **Important:** This class isn’t available in version 32.0 and later. In version 32.0 and later, ConnectApi.EnhancedLinkCapability is used.

Subclass of ConnectApi.FeedItemAttachment.

<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>String</td>
<td>An optional description with a 500 character limit.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>icon</td>
<td>ConnectApi.Icon</td>
<td>An optional icon.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>linkRecordId</td>
<td>String</td>
<td>If linkURL refers to a Salesforce record, linkRecordId contains the ID of the record.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>linkUrl</td>
<td>String</td>
<td>An optional URL to a detail page if there is more content that can’t be displayed inline. Do not specify linkUrl unless you specify a title.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>An optional title to the detail page. If linkUrl is specified, the title links to linkUrl.</td>
<td>28.0–31.0</td>
</tr>
</tbody>
</table>

### ConnectApi.CanvasTemplateAttachment

Attachments in feed items with type CanvasPost.

⚠️ **Important:** This class isn’t available in version 32.0 and later. In version 32.0 and later, ConnectApi.CanvasCapability is used.

Subclass of ConnectApi.FeedItemAttachment.
<table>
<thead>
<tr>
<th>Property</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>description</td>
<td>String</td>
<td>Optional. Description of the canvas app. The maximum length of this field is 500 characters.</td>
<td>29.0–31.0</td>
</tr>
<tr>
<td>developerName</td>
<td>String</td>
<td>Specifies the developer name (API name) of the canvas app.</td>
<td>29.0–31.0</td>
</tr>
<tr>
<td>height</td>
<td>String</td>
<td>Optional. The height of the canvas app in pixels. Default height is 200 pixels.</td>
<td>29.0–31.0</td>
</tr>
<tr>
<td>icon</td>
<td>ConnectApi.Icon</td>
<td>The canvas app icon.</td>
<td>29.0–31.0</td>
</tr>
<tr>
<td>namespacePrefix</td>
<td>String</td>
<td>Optional. The namespace prefix of the Developer Edition organization in which the canvas app was created.</td>
<td>29.0–31.0</td>
</tr>
<tr>
<td>parameters</td>
<td>String</td>
<td>Optional. Parameters passed to the canvas app in JSON format. Example:</td>
<td>29.0–31.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>{'isUpdated': 'true'}</td>
<td></td>
</tr>
<tr>
<td>thumbnailUrl</td>
<td>String</td>
<td>Optional. A URL to a thumbnail image for the canvas app. Maximum dimensions are 120x120 pixels.</td>
<td>29.0–31.0</td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>Specifies the title of the link used to call the canvas app.</td>
<td>29.0–31.0</td>
</tr>
</tbody>
</table>

**ConnectApi.CaseComment**

Attachments in feed items with type CaseCommentPost.

**Important:** This class isn’t available in version 32.0 and later. In version 32.0 and later, `ConnectApi.CaseCommentCapability` is used.

Subclass of `ConnectApi.FeedItemAttachment`.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>actorType</td>
<td>ConnectApi.CaseActorType Enum</td>
<td>Type of user who made the comment.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Customer—if a Chatter customer made the comment</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• CustomerService—if a service representative made the comment</td>
<td></td>
</tr>
<tr>
<td>createdBy</td>
<td>ConnectApi.User Summary</td>
<td>Comment’s creator</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>createdDate</td>
<td>Datetime</td>
<td>ISO 8601 date string, for example, 2011-02-25T18:24:31.000Z</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>Comment’s 18–character ID</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>published</td>
<td>Boolean</td>
<td>Specifies whether the comment has been published</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>text</td>
<td>String</td>
<td>Comment’s text</td>
<td>28.0–31.0</td>
</tr>
</tbody>
</table>
**ConnectApi.ContentAttachment**

Attachments in feed items with the type `ContentPost`.

**Important:** This class isn’t available in version 32.0 and later. In version 32.0 and later, `ConnectApi.ContentCapability` is used.

Subclass of `ConnectApi.FeedItemAttachment`.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>checkSum</td>
<td>String</td>
<td>MD5 checksum for the file.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>contentUrl</td>
<td>String</td>
<td>URL for link files and Google Docs; otherwise the value is <strong>null</strong>.</td>
<td>31.0–31.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>Description of the attachment.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>downloadUrl</td>
<td>String</td>
<td>File’s URL. This value is <strong>null</strong> if the content is a link or a Google Doc.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>fileExtension</td>
<td>String</td>
<td>File’s extension.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>fileSize</td>
<td>String</td>
<td>Size of the file in bytes. If size cannot be determined, returns unknown.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>fileType</td>
<td>String</td>
<td>Type of file.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>hasImagePreview</td>
<td>Boolean</td>
<td><code>true</code> if the file has a preview image available, otherwise <code>false</code>.</td>
<td>28.0–29.0</td>
</tr>
<tr>
<td>hasPdfPreview</td>
<td>Boolean</td>
<td><code>true</code> if the file has a PDF preview available, otherwise <code>false</code>.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>Content’s 18-character ID.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>isInMyFileSync</td>
<td>Boolean</td>
<td><code>true</code> if the file is synced with Salesforce Files Sync; otherwise <code>false</code>.</td>
<td>28.0–31.0</td>
</tr>
</tbody>
</table>

**Note:** Salesforce Files Sync was retired on May 25, 2018.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>mimeType</td>
<td>String</td>
<td>File’s MIME type.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>renditionUrl</td>
<td>String</td>
<td>URL to the file’s rendition resource.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>renditionUrl240By180</td>
<td>String</td>
<td>URL to the 240 x 180 rendition resource for the file. For shared files, renditions process asynchronously after upload. For private files, renditions process when the first file preview is requested, and aren’t available immediately after the file is uploaded.</td>
<td>30.0–31.0</td>
</tr>
<tr>
<td>renditionUrl720By480</td>
<td>String</td>
<td>URL to the 720 x 480 rendition resource for the file. For shared files, renditions process asynchronously after upload. For private files, renditions process when the first file preview is requested, and aren’t available immediately after the file is uploaded.</td>
<td>30.0–31.0</td>
</tr>
<tr>
<td>textPreview</td>
<td>String</td>
<td>Text preview of the file if available; null otherwise.</td>
<td>30.0–31.0</td>
</tr>
</tbody>
</table>
| thumb120By90RenditionStatus| String | Specifies the rendering status of the 120 x 90 preview image of the file. One of these values:  
• Processing—Image is being rendered.  
• Failed—Rendering process failed.  
• Success—Rendering process was successful. | 30.0–31.0         |
Available Version | Description | TypeName
--- | --- | ---
• Na—Rendering is not available for this image.

### thumb240By180 RenditionStatus
- **String**
- Specifies the rendering status of the 240 x 180 preview image of the file. One of these values:
  - Processing—Image is being rendered.
  - Failed—Rendering process failed.
  - Success—Rendering process was successful.
  - Na—Rendering is not available for this image.

### thumb720By480 RenditionStatus
- **String**
- Specifies the rendering status of the 720 x 480 preview image of the file. One of these values:
  - Processing—Image is being rendered.
  - Failed—Rendering process failed.
  - Success—Rendering process was successful.
  - Na—Rendering is not available for this image.

### title
- **String**
- Title of the file.

### versionId
- **String**
- 18-character ID for this version of the content.

---

**ConnectApi.DashboardComponentAttachment**

Attachments in feed items with type DashboardSnapshot.

⚠️ **Important**: This class isn’t available in version 32.0 and later. In version 32.0 and later, ConnectApi.DashboardComponentSnapshotCapability is used.

Subclass of ConnectApi.FeedItemAttachment.

### Name | Type | Description | Available Version
--- | --- | --- | ---
componentId | **String** | Component’s 18–character ID. | 28.0–31.0
componentName | **String** | Name of the component. If no name is saved with the component, returns the localized string, “Untitled Component.”. | 28.0–31.0
dashboardBodyText | **String** | Text displayed next to the actor in the body of a feed item. This is used instead of the default body text. If no text is specified, and there is no default body text, returns null. | 28.0–31.0
dashboardId | **String** | Dashboard’s 18–character ID. | 28.0–31.0
dashboardName | **String** | Name of the dashboard. | 28.0–31.0
fullSizeImageUrl | **String** | URL of the full-sized dashboard image. | 28.0–31.0
lastRefreshDate | **Datetime** | ISO8601 date string, for example, 2011-02-25T18:24:31.000Z, specifying when this dashboard was last refreshed. | 28.0–31.0
### ConnectApi.DatacloudCompany

Information about a Data.com company.

All company information is visible for companies that you purchased and own. If you haven’t purchased a company, some of the fields are hidden. Hidden fields are fully or partially hidden by asterisks “***.”

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>activeContacts</td>
<td>Integer</td>
<td>The number of active Data.com contacts who work in the company.</td>
<td>32.0</td>
</tr>
<tr>
<td>address</td>
<td>ConnectApi.Address</td>
<td>The postal address for the company. This is typically a physical address that can include the city, state, street, and postal code.</td>
<td>32.0</td>
</tr>
<tr>
<td>annualRevenue</td>
<td>Double</td>
<td>The amount of money that the company makes in one year. Annual revenue is measured in US dollars.</td>
<td>32.0</td>
</tr>
<tr>
<td>companyId</td>
<td>String</td>
<td>A unique numerical identifier for the company. This is the Data.com identifier for a company.</td>
<td>32.0</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>A brief synopsis of the company that provides a general overview of the company and what it does.</td>
<td>32.0</td>
</tr>
<tr>
<td>dunsNumber</td>
<td>String</td>
<td>A randomly generated nine-digit number that’s assigned by Dun &amp; Bradstreet (D&amp;B) to identify unique business establishments.</td>
<td>32.0</td>
</tr>
<tr>
<td>industry</td>
<td>String</td>
<td>A description of the type of industry such as “Telecommunications,” “Agriculture,” or “Electronics.”</td>
<td>32.0</td>
</tr>
<tr>
<td>isInactive</td>
<td>Boolean</td>
<td>Indicates whether this company is active (true) or not (false). Inactive companies have out-of-date information in Data.com.</td>
<td>32.0</td>
</tr>
<tr>
<td>isOwned</td>
<td>Boolean</td>
<td>• True: You or your organization owns this company.</td>
<td>32.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>naicsCode</td>
<td>String</td>
<td>North American Industry Classification System (NAICS) codes were created to provide more details about a business's service orientation. The code descriptions are focused on what a business does.</td>
<td>32.0</td>
</tr>
<tr>
<td>naicsDescription</td>
<td>String</td>
<td>A description of the NAICS classification.</td>
<td>32.0</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>The name of the company.</td>
<td>32.0</td>
</tr>
<tr>
<td>numberOfEmployees</td>
<td>Integer</td>
<td>The number of employees who are working for the company.</td>
<td>32.0</td>
</tr>
<tr>
<td>ownership</td>
<td>String</td>
<td>The type of ownership of the company:</td>
<td>32.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Public</td>
<td>32.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Private</td>
<td>32.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Government</td>
<td>32.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Other</td>
<td>32.0</td>
</tr>
<tr>
<td>phoneNumbers</td>
<td>ConnectApi.PhoneNumber</td>
<td>The list of telephone numbers for the company, including the type. Here are some possible types of telephone numbers.</td>
<td>32.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Mobile</td>
<td>32.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Work</td>
<td>32.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Fax</td>
<td>32.0</td>
</tr>
<tr>
<td>sic</td>
<td>String</td>
<td>Standard Industrial Codes (SIC) is a numbering convention that indicates what type of service a business provides. It's a four-digit value.</td>
<td>32.0</td>
</tr>
<tr>
<td>sicDescription</td>
<td>String</td>
<td>A description of the SIC classification.</td>
<td>32.0</td>
</tr>
<tr>
<td>site</td>
<td>String</td>
<td>Company's site. For example, HQ, Single Location, or Branch. An organization status of the company.</td>
<td>32.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Branch: a secondary location to a headquarter location.</td>
<td>32.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Headquarter: the parent company has branches or subsidiaries.</td>
<td>32.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Single Location: a single business with no subsidiaries or branches.</td>
<td>32.0</td>
</tr>
</tbody>
</table>
### ConnectApi.DatacloudCompanies

Lists all companies that were purchased in a specific order, page URLs, and the number of companies in the order.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>companies</td>
<td>ConnectApi.DatacloudCompany</td>
<td>A detailed list of companies that were part of a single order.</td>
<td>32.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>The URL for the current page of a list of companies.</td>
<td>32.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or <code>null</code> if there isn’t a next page.</td>
<td>32.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>The URL to the previous page of companies that were viewed before the current page. If this value is <code>null</code>, there’s no previous page.</td>
<td>32.0</td>
</tr>
<tr>
<td>total</td>
<td>Integer</td>
<td>The number of companies in the order. You can calculate the number of pages to display by dividing this number by your page size. The default page size is 25.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

### ConnectApi.DatacloudContact

Information about a Data.com contact.

All contact information is visible for contacts that you purchased. If you have not purchased a contact, some of the fields will be hidden. Hidden fields are fully or partially hidden by asterisks `***`.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>address</td>
<td>ConnectApi.Address</td>
<td>The contact’s business address.</td>
<td>32.0</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>---------------</td>
<td>------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>companyId</td>
<td>String</td>
<td>A unique numerical identifier for the company where the contact works. This is the Data.com identifier for a company.</td>
<td>32.0</td>
</tr>
<tr>
<td>companyName</td>
<td>String</td>
<td>The company name where the contact works.</td>
<td>32.0</td>
</tr>
<tr>
<td>contactId</td>
<td>String</td>
<td>A unique numerical identifier for the contact. This is the Data.com identifier for a contact.</td>
<td>32.0</td>
</tr>
<tr>
<td>department</td>
<td>String</td>
<td>The department in the company where the contact works. Here are some possible departments.</td>
<td>32.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Engineering</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• IT</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Marketing</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Sales</td>
<td></td>
</tr>
<tr>
<td>email</td>
<td>String</td>
<td>The most current business email address for the contact.</td>
<td>32.0</td>
</tr>
<tr>
<td>firstName</td>
<td>String</td>
<td>The first name of the contact.</td>
<td>32.0</td>
</tr>
<tr>
<td>isInactive</td>
<td>Boolean</td>
<td>Whether this contact is active (true) or not (false). Inactive contacts have out-of-date information in Data.com.</td>
<td>32.0</td>
</tr>
<tr>
<td>isOwned</td>
<td>Boolean</td>
<td>Whether this contact is owned (true) or not (false).</td>
<td>32.0</td>
</tr>
<tr>
<td>lastName</td>
<td>String</td>
<td>The last name of the contact.</td>
<td>32.0</td>
</tr>
<tr>
<td>level</td>
<td>String</td>
<td>A human resource label that designates a person’s level in the company. Here are some possible levels.</td>
<td>32.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• C-Level</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Director</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Manager</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Staff</td>
<td></td>
</tr>
<tr>
<td>phoneNumbers</td>
<td>ConnectApi.PhoneNumber</td>
<td>Telephone numbers for the contact, which can include direct-dial business telephone numbers, mobile telephone numbers, and business headquarters telephone numbers. The type of telephone number is also indicated.</td>
<td>32.0</td>
</tr>
<tr>
<td>title</td>
<td>String</td>
<td>The title of the contact, such as CEO or Vice President.</td>
<td>32.0</td>
</tr>
<tr>
<td>updatedDate</td>
<td>Datetime</td>
<td>The date of the most recent change to this contact’s information.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

SEE ALSO:
ConnectApi.DatacloudContacts
### ConnectApi.DatacloudContacts

Lists all contacts that were purchased in the specific order, page URLs, and the number of contacts in the order.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>contacts</td>
<td>List&lt;ConnectApi.DatacloudContact&gt;</td>
<td>A detailed list of purchased contacts.</td>
<td>32.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>URL to the current page of contacts.</td>
<td>32.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or null if there isn’t a next page.</td>
<td>32.0</td>
</tr>
<tr>
<td>previousPageUrl</td>
<td>String</td>
<td>URL to the previous page of contacts. This value is null if there is no previous page.</td>
<td>32.0</td>
</tr>
<tr>
<td>total</td>
<td>Integer</td>
<td>Number of contacts that are associated with this order. Can be greater than the number of contacts that are shown on a single page.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

### ConnectApi.DatacloudOrder

Represents a Datacloud order.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>entityUrl</td>
<td>String</td>
<td>URL to a list of contacts or companies that were purchased with this order.</td>
<td>32.0</td>
</tr>
<tr>
<td>id</td>
<td>String</td>
<td>Unique number that’s used to track your order information.</td>
<td>32.0</td>
</tr>
<tr>
<td>purchaseCount</td>
<td>Integer</td>
<td>Number of contacts or companies that were purchased for this order.</td>
<td>32.0</td>
</tr>
<tr>
<td>purchaseDate</td>
<td>Datetime</td>
<td>Purchase date for this order.</td>
<td>32.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>GET request URL for this order.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

### ConnectApi.DatacloudPurchaseUsage

Information about Data.com point usage for monthly and list pool users.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>listpoolCreditsAvailable</td>
<td>Integer</td>
<td>The points or credits that are available in a pool of credits for your organization. This pool of credits can be used by any List Pool User in your organization.</td>
<td>32.0</td>
</tr>
<tr>
<td>listpoolCreditsUsed</td>
<td>Integer</td>
<td>The points or credits that have been used from a pool of credits that are used by List Pool Users to purchase records.</td>
<td>32.0</td>
</tr>
</tbody>
</table>
### ConnectApi.EmailMessage

Email message from a case.

**Important:** This class isn’t available in version 32.0 and later. In version 32.0 and later, `ConnectApi.EmailMessageCapability` is used.

Subclass of `ConnectApi.FeedItemAttachment`.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
</table>
| direction          | ConnectApi.EmailMessageDirection Enum | The direction of the email message.  
  - Inbound—An inbound message (sent by a customer).  
  - Outbound—An outbound message (sent to a customer by a support agent). | 29.0–31.0         |
| emailMessageId     | String                | The ID of the email message.                                                | 29.0–31.0         |
| subject            | String                | The subject of the email message.                                           | 29.0–31.0         |
| textBody           | String                | The body of the email message.                                              | 29.0–31.0         |
| toAddresses        | List<ConnectApi.EmailAddress> | A list of email addresses to send the message to.                          | 29.0–31.0         |

### ConnectApi.FeedItemAttachment

Feed item attachment.

**Important:** This class isn’t available in version 32.0 and later. In version 32.0 and later, `ConnectApi.FeedElementCapability` is used.

This class is abstract.

Subclasses:
- `ConnectApi.ApprovalAttachment`
- `ConnectApi.BasicTemplateAttachment`
- `ConnectApi.CanvasTemplateAttachment`
- `ConnectApi.EmailMessage`
- `ConnectApi.CaseComment`
- `ConnectApi.ContentAttachment`
- `ConnectApi.DashboardComponentAttachment`
• ConnectApi.FeedPoll
• ConnectApi.LinkAttachment
• ConnectApi.RecordSnapshotAttachment
• ConnectApi.TrackedChangeAttachment

Message segments in a feed item are typed as ConnectApi.MessageSegment. Feed item capabilities are typed as ConnectApi.FeedItemCapability. Record fields are typed as ConnectApi.AbstractRecordField. These classes are all abstract and have several concrete subclasses. At runtime you can use instanceof to check the concrete types of these objects and then safely proceed with the corresponding downcast. When you downcast, you must have a default case that handles unknown subclasses.

⚠️ Important: The composition of a feed can change between releases. Write your code to handle instances of unknown subclasses.

ConnectApi.FeedItemPage
A paged collection of ConnectApi.FeedItem objects.

⚠️ Important: This class isn’t available in version 32.0 and later. In version 32.0 and later, ConnectApi.FeedElementPage is used.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>currentPageToken</td>
<td>String</td>
<td>Token identifying the current page.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>currentPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the current page.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>isModifiedToken</td>
<td>String</td>
<td>Opaque polling token to use in the since parameter of the ChatterFeeds.isModified method. The token describes when the feed was last modified.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>⚠️ Important: This feature is available through a Feed Polling pilot program. This pilot program is closed and not accepting new participants.</td>
<td></td>
</tr>
<tr>
<td>isModifiedUrl</td>
<td>String</td>
<td>Connect REST API URL with a since request parameter that contains an opaque token that describes when the feed was last modified. Returns null if the feed isn’t a news feed. Use this URL to poll a news feed for updates.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>⚠️ Important: This feature is available through a Feed Polling pilot program. This pilot program is closed and not accepting new participants.</td>
<td></td>
</tr>
<tr>
<td>items</td>
<td>List&lt;ConnectApi.FeedItem&gt;</td>
<td>List of feed items</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>nextPageToken</td>
<td>String</td>
<td>Token identifying the next page, or null if there isn’t a next page.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>nextPageUrl</td>
<td>String</td>
<td>Connect REST API URL identifying the next page, or null if there isn’t a next page.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>updatesToken</td>
<td>String</td>
<td>Token to use in an updatedSince parameter, or null if not available.</td>
<td>30.0–31.0</td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
<td>Available Version</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>updatesUrl</td>
<td>String</td>
<td>A Connect REST API resource with a query string containing the value of the <code>updatesToken</code> property. The resource returns the feed items that have been updated since the last request. Use the URL as it is—do not modify it. Property is <code>null</code> if not available.</td>
<td>30.0–31.0</td>
</tr>
</tbody>
</table>

**ConnectApi.FeedItemTopicPage**

Feed item topic page.

**Important:** This class isn’t available in version 32.0 and later. In version 32.0 and later, `ConnectApi.TopicsCapability` is used.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>canAssignTopics</td>
<td>Boolean</td>
<td>true if a topic can be assigned to the feed item, false otherwise.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>topics</td>
<td>List&lt;ConnectApi.Topic&gt;</td>
<td>List of topics.</td>
<td>28.0–31.0</td>
</tr>
</tbody>
</table>

**ConnectApi.FeedPoll**

Attachment of `ConnectApi.FeedItem` objects where the `type` property is `PollPost`.

**Important:** This class isn’t available in version 32.0 and later. In version 32.0 and later, `ConnectApi.PollCapability` is used.

Subclass of `ConnectApi.FeedItemAttachment`.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>choices</td>
<td>List&lt;ConnectApi.FeedPollChoice&gt;</td>
<td>List of choices for poll.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>myChoiceId</td>
<td>String</td>
<td>ID of the poll choice that the context user has voted for in this poll. Returns <code>null</code> if the context user hasn’t voted.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>totalVoteCount</td>
<td>Integer</td>
<td>Total number of votes cast on the feed poll item.</td>
<td>28.0–31.0</td>
</tr>
</tbody>
</table>

**ConnectApi.LinkAttachment**

Link attached to a feed item.

**Important:** This class isn’t available in version 32.0 and later. In version 32.0 and later, `ConnectApi.LinkCapability` is used.

Subclass of `ConnectApi.FeedItemAttachment`.
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>title</td>
<td>String</td>
<td>Title given to the link if available, otherwise, null.</td>
<td>28.0–31.0</td>
</tr>
<tr>
<td>url</td>
<td>String</td>
<td>The link URL.</td>
<td>28.0–31.0</td>
</tr>
</tbody>
</table>

**ConnectApi.NonEntityRecommendation**

A recommendation for a non-Salesforce entity, such as an application.


⚠️ **Important:** ConnectApi.NonEntityRecommendation isn’t used in version 34.0 and later. In version 34.0 and later, ConnectApi.EntityRecommendation is used for all recommendations.

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>displayLabel</td>
<td>String</td>
<td>Localized label of the non-entity object.</td>
<td>32.0</td>
</tr>
<tr>
<td>motif</td>
<td>ConnectApi.Motif</td>
<td>Motif for the non-entity object.</td>
<td>32.0</td>
</tr>
</tbody>
</table>

**ConnectApi.RecordSnapshotAttachment**

Fields of a record at the point in time when the record was created.

⚠️ **Important:** This class isn’t available in version 32.0 and later. In version 32.0 and later, ConnectApi.RecordSnapshotCapability is used.

Subclass of ConnectApi.FeedItemAttachment.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>recordView</td>
<td>ConnectApi.RecordView</td>
<td>The representation of the record.</td>
<td>29.0–31.0</td>
</tr>
</tbody>
</table>

**ConnectApi.TrackedChangeAttachment**

Tracked change attachment to a feed item.

⚠️ **Important:** This class isn’t available in version 32.0 and later. In version 32.0 and later, ConnectApi.TrackedChangesCapability is used.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Available Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>changes</td>
<td>List&lt;ConnectApi.TrackedChangeItem&gt;</td>
<td>A list of tracked changes.</td>
<td>28.0–31.0</td>
</tr>
</tbody>
</table>
ConnectApi Enums

 Enums specific to the ConnectApi namespace.

ConnectApi enums inherit all properties and methods of Apex enums.

 Enums are not versioned. Enum values are returned in all API versions. Clients should handle values they don’t understand gracefully.

<table>
<thead>
<tr>
<th>Enum</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConnectApi.ActionLinkExecutionsAllowed</td>
<td>Number of times an action link can be executed.</td>
</tr>
<tr>
<td>• Once</td>
<td>An action link can be executed only one time across all users.</td>
</tr>
<tr>
<td>• OncePerUser</td>
<td>An action link can be executed only one time for each user.</td>
</tr>
<tr>
<td>• Unlimited</td>
<td>An action link can be executed an unlimited number of times by each user. If the action link’s actionType is Api or ApiAsync, you can’t use this value.</td>
</tr>
<tr>
<td>ConnectApi.ActionLinkType</td>
<td>Type of action link.</td>
</tr>
<tr>
<td>• Api</td>
<td>The action link calls a synchronous API at the action URL. Salesforce sets the status to SuccessfulStatus or FailedStatus based on the HTTP status code returned by your server.</td>
</tr>
<tr>
<td>• ApiAsync</td>
<td>The action link calls an asynchronous API at the action URL. The action remains in a PendingStatus state until a third party makes a request to /connect/action-links/actionLinkId to set the status to SuccessfulStatus or FailedStatus when the asynchronous operation is complete.</td>
</tr>
<tr>
<td>• Download</td>
<td>The action link downloads a file from the action URL.</td>
</tr>
<tr>
<td>• Ui</td>
<td>The action link takes the user to a web page at the action URL.</td>
</tr>
<tr>
<td>ConnectApi.ActivitySharingTypeEnum</td>
<td>Type of sharing operation.</td>
</tr>
<tr>
<td>• Everyone</td>
<td>The activity is shared with everyone.</td>
</tr>
<tr>
<td>• MyGroups</td>
<td>The activity is shared only with a selection of the context user’s groups.</td>
</tr>
<tr>
<td>• OnlyMe</td>
<td>The activity is private.</td>
</tr>
<tr>
<td>ConnectApi.AdjustmentAmountScope</td>
<td>Scope of the price adjustment amount.</td>
</tr>
<tr>
<td>• Total</td>
<td>The adjustment scope is the total price.</td>
</tr>
<tr>
<td>• Unit</td>
<td>The adjustment scope is the unit price.</td>
</tr>
<tr>
<td>• UnproratedTotal</td>
<td>The adjustment scope is the unprorated total price.</td>
</tr>
<tr>
<td>ConnectApi.AdjustmentType</td>
<td>How the price adjustment amount is calculated.</td>
</tr>
<tr>
<td>• AdjustmentAmount</td>
<td>The adjustment is a fixed amount.</td>
</tr>
<tr>
<td>• AdjustmentPercentage</td>
<td>The adjustment is a percentage.</td>
</tr>
<tr>
<td>ConnectApi.ArticleTopicJobType</td>
<td>Type of operation to perform on articles and topics.</td>
</tr>
<tr>
<td>• AssignTopicsToArticle</td>
<td>Assign topics to articles in a data category.</td>
</tr>
<tr>
<td>• UnassignTopicsFromArticle</td>
<td>Unassign topics from articles in a data category.</td>
</tr>
<tr>
<td>Enum</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ConnectApi.AsyncOperationStatus</td>
<td>Asynchronous processing status of the cart, if asynchronous processing is enabled for the store.</td>
</tr>
<tr>
<td></td>
<td>• Completed</td>
</tr>
<tr>
<td></td>
<td>•Errored</td>
</tr>
<tr>
<td></td>
<td>• Processing</td>
</tr>
<tr>
<td>ConnectApi.AudienceCriteriaOperator</td>
<td>Operator used in the personalization audience criterion.</td>
</tr>
<tr>
<td></td>
<td>• Contains</td>
</tr>
<tr>
<td></td>
<td>• Equal</td>
</tr>
<tr>
<td></td>
<td>• GreaterThan</td>
</tr>
<tr>
<td></td>
<td>• GreaterThanOrEqual</td>
</tr>
<tr>
<td></td>
<td>• Includes</td>
</tr>
<tr>
<td></td>
<td>• LessThan</td>
</tr>
<tr>
<td></td>
<td>• LessThanOrEqual</td>
</tr>
<tr>
<td></td>
<td>• NotEqual</td>
</tr>
<tr>
<td></td>
<td>• NotIncludes</td>
</tr>
<tr>
<td></td>
<td>• StartsWith</td>
</tr>
<tr>
<td>ConnectApi.AudienceCriteriaType</td>
<td>Type of personalization audience criterion.</td>
</tr>
<tr>
<td></td>
<td>• Audience—Criterion based on audience.</td>
</tr>
<tr>
<td></td>
<td>• Default—Audience has no criteria.</td>
</tr>
<tr>
<td></td>
<td>• Domain—Criterion based on domain.</td>
</tr>
<tr>
<td></td>
<td>• FieldBased—Criterion based on object fields.</td>
</tr>
<tr>
<td></td>
<td>• GeoLocation—Criterion based on location.</td>
</tr>
<tr>
<td></td>
<td>• Permission—Criterion based on standard or custom permissions.</td>
</tr>
<tr>
<td></td>
<td>• Profile—Criterion based on profile.</td>
</tr>
<tr>
<td>ConnectApi.BannerStyle</td>
<td>Decorates a feed item with a color and set of icons.</td>
</tr>
<tr>
<td></td>
<td>• Announcement—An announcement displays in a designated location in the Salesforce UI until 11:59 p.m. on its expiration date, unless it’s deleted or replaced by another announcement.</td>
</tr>
<tr>
<td>ConnectApi.BillingFrequency</td>
<td>Reserved for future use.</td>
</tr>
<tr>
<td></td>
<td>• Active</td>
</tr>
<tr>
<td></td>
<td>• Inactive</td>
</tr>
<tr>
<td>ConnectApi.BundleType</td>
<td>Type of bundle.</td>
</tr>
<tr>
<td></td>
<td>• GenericBundle—A bundle that contains no additional information and is just a collection of feed elements.</td>
</tr>
<tr>
<td>Enum</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><a href="#">TrackedChanges</a></td>
<td>A bundle that represents a collection of feed-tracked changes. The bundle includes summary information about the feed-tracked changes that make up the bundle.</td>
</tr>
<tr>
<td><a href="#">DefinitionType</a></td>
<td>Definition type of the calculated insight.</td>
</tr>
<tr>
<td><a href="#">CalculatedMetric</a></td>
<td>Type of tax calculation.</td>
</tr>
<tr>
<td><a href="#">Actual</a></td>
<td>Calculated tax represents the final taxed amount for the transaction.</td>
</tr>
<tr>
<td><a href="#">Estimated</a></td>
<td>Calculated tax represents only an estimated value before the transaction is finalized.</td>
</tr>
<tr>
<td><a href="#">Disabled</a></td>
<td>Indicates whether a named credential is enabled for callout.</td>
</tr>
<tr>
<td><a href="#">Enabled</a></td>
<td></td>
</tr>
<tr>
<td><a href="#">CreditCard</a></td>
<td>Indicates a credit card or debit card.</td>
</tr>
<tr>
<td><a href="#">DebitCard</a></td>
<td></td>
</tr>
<tr>
<td><a href="#">AmericanExpress</a></td>
<td>Credit card issuer.</td>
</tr>
<tr>
<td><a href="#">DinersClub</a></td>
<td></td>
</tr>
<tr>
<td><a href="#">JCB</a></td>
<td></td>
</tr>
<tr>
<td><a href="#">Maestro</a></td>
<td></td>
</tr>
<tr>
<td><a href="#">MasterCard</a></td>
<td></td>
</tr>
<tr>
<td><a href="#">Visa</a></td>
<td></td>
</tr>
<tr>
<td><a href="#">CreatedDateAsc</a></td>
<td>Sort order for items in a cart.</td>
</tr>
<tr>
<td><a href="#">CreatedDateDesc</a></td>
<td></td>
</tr>
<tr>
<td><a href="#">NameAsc</a></td>
<td></td>
</tr>
<tr>
<td><a href="#">NameDesc</a></td>
<td></td>
</tr>
<tr>
<td><a href="#">SalesPriceAsc</a></td>
<td></td>
</tr>
<tr>
<td><a href="#">SalesPriceDesc</a></td>
<td></td>
</tr>
<tr>
<td><a href="#">DeliveryCharge</a></td>
<td>Type of item in a cart.</td>
</tr>
<tr>
<td><a href="#">Product</a></td>
<td></td>
</tr>
<tr>
<td>Enum</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ConnectApi. CartPromotionType</td>
<td>Level of the promotion target.</td>
</tr>
<tr>
<td>ConnectApi. CartStatus</td>
<td>Status of the cart.</td>
</tr>
<tr>
<td>ConnectApi. CartTaxType</td>
<td>Tax type of the cart.</td>
</tr>
<tr>
<td>ConnectApi. CartType</td>
<td>Type of cart.</td>
</tr>
<tr>
<td>ConnectApi. CaseActorType</td>
<td>Type of user who made the comment.</td>
</tr>
<tr>
<td>ConnectApi. CaseCommentEventType</td>
<td>Event type of a comment in Case Feed.</td>
</tr>
</tbody>
</table>

- **ConnectApi. CartMessageSeverity**
  - Error
  - Info
  - Warning

- **ConnectApi. CartPromotionType**
  - Cart—The target is cart-level.
  - Item—The target is item-level.

- **ConnectApi. CartStatus**
  - Active—Cart is created and available for modifications, like adding or removing products or promotions.
  - Checkout—Cart is in checkout. If the customer modifies the cart, the current checkout session is canceled.
  - Closed—Checkout is complete and an order was created. The cart cannot be modified.
  - PendingClosed—Cart is marked to be closed, but the request isn’t completed yet. The cart can’t be modified. This value is available in API version 57.0 and later.
  - PendingDelete—Cart is marked for delete, but the request isn’t completed yet. The cart can’t be modified.
  - Processing—Cart is processing. For example, taxes are being calculated. The cart can’t be modified.

- **ConnectApi. CartTaxType**
  - Gross—Gross taxation policy.
  - Net—Net taxation policy.

- **ConnectApi. CartType**
  - Cart—Cart created by a customer.
  - ReadOnly—Clone of a Template cart that the customer can check out with.
  - Template—Cart created by an internal user.

- **ConnectApi. CaseActorType**
  - Customer—if a Chatter customer made the comment
  - CustomerService—if a service representative made the comment

- **ConnectApi. CaseCommentEventType**
  - NewInternal—A case comment that has newly been marked Internal Only.
  - NewPublished—A newly published case comment.
  - NewPublishedByCustomer—A case comment by a customer that was newly published.
  - PublishExisting—An existing case comment that was republished.
### Enum

<table>
<thead>
<tr>
<th>Enum</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConnectApi.CdpIdentityResolution</td>
<td>Source object for an identity resolution ruleset.</td>
</tr>
<tr>
<td>ConfigurationType</td>
<td>• Account</td>
</tr>
<tr>
<td></td>
<td>• Individual</td>
</tr>
<tr>
<td>ConnectApi.CdpIdentityResolution</td>
<td>Match method for a match rule criterion.</td>
</tr>
<tr>
<td>MatchMethodType</td>
<td>• Exact—Exact match.</td>
</tr>
<tr>
<td></td>
<td>• ExactNormalized—Exact normalized match.</td>
</tr>
<tr>
<td></td>
<td>• Fuzzy—Fuzzy match with medium precision.</td>
</tr>
<tr>
<td></td>
<td>• FuzzyHigh—Fuzzy match with high precision.</td>
</tr>
<tr>
<td></td>
<td>• FuzzyLow—Fuzzy match with low precision.</td>
</tr>
<tr>
<td>ConnectApi.CdpIdentityResolution</td>
<td>Default reconciliation rule applied to fields in the object the reconciliation rule applies to.</td>
</tr>
<tr>
<td>ReconciliationRuleType</td>
<td>• LastUpdated</td>
</tr>
<tr>
<td></td>
<td>• MostFrequent</td>
</tr>
<tr>
<td></td>
<td>• SourceSequence</td>
</tr>
<tr>
<td>ConnectApi.CdpIdentityResolution</td>
<td>Result of an identity resolution ruleset job run.</td>
</tr>
<tr>
<td>RunNowResultCode</td>
<td>• ExceededMaximumNumberOfSuccessfulRunsAllowedIn24Hours</td>
</tr>
<tr>
<td></td>
<td>• IdentityResolutionJobIsAlreadyRunning</td>
</tr>
<tr>
<td></td>
<td>• NoPendingChangesJobRunSkipped</td>
</tr>
<tr>
<td></td>
<td>• SuccessfullySubmittedIdentityResolutionJobRunRequest</td>
</tr>
<tr>
<td>ConnectApi.CommentType</td>
<td>Type of comment.</td>
</tr>
<tr>
<td></td>
<td>• ContentComment—Comment holds a content capability.</td>
</tr>
<tr>
<td></td>
<td>• TextComment—Comment contains only text.</td>
</tr>
<tr>
<td>ConnectApi.CommerceAddressSort</td>
<td>Sort order for Commerce addresses.</td>
</tr>
<tr>
<td></td>
<td>• CreatedDateAsc—Sort in ascending order of created date.</td>
</tr>
<tr>
<td></td>
<td>• CreatedDateDesc—Sort in descending order of created date.</td>
</tr>
<tr>
<td></td>
<td>• NameAsc—Sort in ascending order of name.</td>
</tr>
<tr>
<td></td>
<td>• NameDesc—Sort in descending order of name.</td>
</tr>
<tr>
<td>Enum</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>ConnectApi.CommerceSearchAttributeType</td>
<td>Search attribute type.</td>
</tr>
<tr>
<td>• Custom</td>
<td></td>
</tr>
<tr>
<td>• ProductAttribute</td>
<td></td>
</tr>
<tr>
<td>• ProductCategory</td>
<td></td>
</tr>
<tr>
<td>• Product2</td>
<td></td>
</tr>
<tr>
<td>• Standard</td>
<td></td>
</tr>
<tr>
<td>ConnectApi.CommerceSearchFacetDisplayType</td>
<td>Display type of the facet.</td>
</tr>
<tr>
<td>• CategoryTree</td>
<td></td>
</tr>
<tr>
<td>• DatePicker</td>
<td></td>
</tr>
<tr>
<td>• MultiSelect</td>
<td></td>
</tr>
<tr>
<td>• SingleSelect</td>
<td></td>
</tr>
<tr>
<td>ConnectApi.CommerceSearchFacetType</td>
<td>Search facet type.</td>
</tr>
<tr>
<td>• DistinctValue</td>
<td></td>
</tr>
<tr>
<td>ConnectApi.CommerceSearchGroupingOption</td>
<td>Grouping option for search results.</td>
</tr>
<tr>
<td>• NoGrouping—Search results aren’t grouped.</td>
<td></td>
</tr>
<tr>
<td>• VariationParent—Search results are grouped by the variation parent.</td>
<td></td>
</tr>
<tr>
<td>ConnectApi.CommerceSearchIndexBuildType</td>
<td>Build type of the index.</td>
</tr>
<tr>
<td>• Full</td>
<td></td>
</tr>
<tr>
<td>• Incremental</td>
<td></td>
</tr>
<tr>
<td>ConnectApi.CommerceSearchIndexCreationType</td>
<td>Creation type of the index.</td>
</tr>
<tr>
<td>• Manual</td>
<td></td>
</tr>
<tr>
<td>• Scheduled</td>
<td></td>
</tr>
<tr>
<td>ConnectApi.CommerceSearchIndexStatus</td>
<td>Status of the index.</td>
</tr>
<tr>
<td>• Completed</td>
<td></td>
</tr>
<tr>
<td>• Failed</td>
<td></td>
</tr>
<tr>
<td>• InProgress</td>
<td></td>
</tr>
<tr>
<td>ConnectApi.CommerceSearchIndexUsage</td>
<td>Usage of the index.</td>
</tr>
<tr>
<td>• Live</td>
<td></td>
</tr>
<tr>
<td>• OutOfUse</td>
<td></td>
</tr>
<tr>
<td>ConnectApi.CommerceSearchSortRuleDirection</td>
<td>Direction of the sort rule.</td>
</tr>
<tr>
<td>• Ascending—Sorts in ascending alphanumeric order (A–Z, 0–9).</td>
<td></td>
</tr>
<tr>
<td>• Default—When no direction is defined, sorts by relevance.</td>
<td></td>
</tr>
<tr>
<td>• Descending—Sorts in descending alphanumeric order (Z–A, 9–0).</td>
<td></td>
</tr>
<tr>
<td>Enum</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>ConnectApi. CommerceSearchSortRule LabelSuffix</td>
<td>Label suffix of the sort rule.</td>
</tr>
<tr>
<td>Ascen—Label suffix for 'Asc'</td>
<td></td>
</tr>
<tr>
<td>Ascending—Label suffix for 'Ascending'</td>
<td></td>
</tr>
<tr>
<td>Az—Label suffix for 'A-Z'</td>
<td></td>
</tr>
<tr>
<td>Desce—Label suffix for 'Desc'</td>
<td></td>
</tr>
<tr>
<td>Descending—Label suffix for 'Descending'</td>
<td></td>
</tr>
<tr>
<td>FewMany—Label suffix for 'Few-Many'</td>
<td></td>
</tr>
<tr>
<td>HeavyLight—Label suffix for 'Heavy-Light'</td>
<td></td>
</tr>
<tr>
<td>HighLow—Label suffix for 'High-Low'</td>
<td></td>
</tr>
<tr>
<td>HighestLowest—Label suffix for 'Highest-Lowest'</td>
<td></td>
</tr>
<tr>
<td>LightHeavy—Label suffix for 'Light-Heavy'</td>
<td></td>
</tr>
<tr>
<td>LowHigh—Label suffix for 'Low-High'</td>
<td></td>
</tr>
<tr>
<td>LowestHighest—Label suffix for 'Lowest-Highest'</td>
<td></td>
</tr>
<tr>
<td>ManyFew—Label suffix for 'Many-Few'</td>
<td></td>
</tr>
<tr>
<td>NewOld—Label suffix for 'New-Old'</td>
<td></td>
</tr>
<tr>
<td>Newest—Label suffix for 'Newest'</td>
<td></td>
</tr>
<tr>
<td>NewestOldest—Label suffix for 'Newest-Oldest'</td>
<td></td>
</tr>
<tr>
<td>NineZero—Label suffix for '9-0'</td>
<td></td>
</tr>
<tr>
<td>OldNew—Label suffix for 'Old-New'</td>
<td></td>
</tr>
<tr>
<td>Oldest—Label suffix for 'Oldest'</td>
<td></td>
</tr>
<tr>
<td>OldestNewest—Label suffix for 'Oldest-Newest'</td>
<td></td>
</tr>
<tr>
<td>PriceDecreasing—Label suffix for '$$-$'</td>
<td></td>
</tr>
<tr>
<td>PriceIncreasing—Label suffix for '$-$$$'</td>
<td></td>
</tr>
<tr>
<td>ThickThin—Label suffix for 'Thick-Thin'</td>
<td></td>
</tr>
<tr>
<td>ThinThick—Label suffix for 'Thin-Thick'</td>
<td></td>
</tr>
<tr>
<td>Za—Label suffix for 'Z-A'</td>
<td></td>
</tr>
<tr>
<td>ZeroNine—Label suffix for '0-9'</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Enum</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConnectApi. CommerceSearchSortRule SortRuleType</td>
<td>Type of sort rule.</td>
</tr>
<tr>
<td>ProductAttributeBased—Sorts by product attribute fields.</td>
<td></td>
</tr>
<tr>
<td>ProductBased—Sorts by product field data.</td>
<td></td>
</tr>
<tr>
<td>Relevancy—Sorts by product and catalog term frequency.</td>
<td></td>
</tr>
<tr>
<td>SortByPricebook—Sorts by product prices defined in the specified pricebook (version 55.0 and later).</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Enum</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConnectApi. CommerceSearchTop ProductType</td>
<td>Type of the top product to return for each product group in search results.</td>
</tr>
<tr>
<td>VariationParent</td>
<td></td>
</tr>
<tr>
<td>Enum</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ConnectApi.CommunityFlagReasonType</td>
<td>Reason a post, comment, or file is flagged.</td>
</tr>
<tr>
<td></td>
<td>• FlaggedByRule—Moderation rule flagged the item.</td>
</tr>
<tr>
<td></td>
<td>• FlaggedBySystem—Einstein flagged the item.</td>
</tr>
<tr>
<td></td>
<td>• FlaggedByUserAsInappropriate—User flagged the item as inappropriate.</td>
</tr>
<tr>
<td></td>
<td>• FlaggedByUserAsSpam—User flagged the item as spam.</td>
</tr>
<tr>
<td>ConnectApi.CommunityFlagType</td>
<td>Type of moderation flag.</td>
</tr>
<tr>
<td></td>
<td>• FlagAsInappropriate—Flag for inappropriate content.</td>
</tr>
<tr>
<td></td>
<td>• FlagAsSpam—Flag for spam.</td>
</tr>
<tr>
<td>ConnectApi.CommunityFlagVisibility</td>
<td>Visibility behavior of a flag for various user types.</td>
</tr>
<tr>
<td></td>
<td>• ModeratorsOnly—The flag is visible only to users with moderation permissions on the flagged element or item.</td>
</tr>
<tr>
<td></td>
<td>• SelfAndModerators—The flag is visible to the creator of the flag and to users with moderation permissions on the flagged element or item.</td>
</tr>
<tr>
<td></td>
<td>• Live</td>
</tr>
<tr>
<td></td>
<td>• Inactive</td>
</tr>
<tr>
<td></td>
<td>• UnderConstruction</td>
</tr>
<tr>
<td>ConnectApi.ActivityType</td>
<td>Type of activity.</td>
</tr>
<tr>
<td></td>
<td>• All</td>
</tr>
<tr>
<td></td>
<td>• Event</td>
</tr>
<tr>
<td></td>
<td>• Task</td>
</tr>
<tr>
<td>ConnectApi.ContentHubAuthenticationProtocol</td>
<td>Authentication protocol used for the repository.</td>
</tr>
<tr>
<td></td>
<td>• NoAuthentication—Repository doesn’t require authentication.</td>
</tr>
<tr>
<td></td>
<td>• Oauth—Repository uses OAuth authentication protocol.</td>
</tr>
<tr>
<td></td>
<td>• Password—Repository uses user name and password authentication protocol.</td>
</tr>
<tr>
<td>ConnectApi.ContentHubDirectoryEntryType</td>
<td>Type of directory entry.</td>
</tr>
<tr>
<td></td>
<td>• GroupEntry</td>
</tr>
<tr>
<td></td>
<td>• UserEntry</td>
</tr>
<tr>
<td>ConnectApi.ContentHubExternalItemSharingType</td>
<td>Sharing status for the external file.</td>
</tr>
<tr>
<td></td>
<td>• DomainSharing—File is shared with the domain.</td>
</tr>
<tr>
<td></td>
<td>• PrivateSharing—File is private or shared only with individuals.</td>
</tr>
<tr>
<td></td>
<td>• PublicSharing—File is publicly shared.</td>
</tr>
<tr>
<td>Enum</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| `ConnectApi.ContentHub.GroupType` | Type of group.  
  - Everybody—Group is public to everybody.  
  - EverybodyInDomain—Group is public to everybody in the same domain.  
  - Unknown—Group type is unknown. |
| `ConnectApi.ContentHub.ItemType` | Item types.  
  - Any—Includes files and folders.  
  - FilesOnly—Includes files only.  
  - FoldersOnly—Includes folders only. |
  - ContentStreamAllowed  
  - ContentStreamNotAllowed  
  - ContentStreamRequired |
| `ConnectApi.ContentHub.VariableType` | Data type of the value of the field.  
  - BooleanType  
  - DateTimeType  
  - DecimalType  
  - HtmlType  
  - IdType  
  - IntegerType  
  - StringType  
  - UriType  
  - XmlType |
  - Api  
  - Slack |
| `ConnectApi.CreateCredentialAction` | Action to take when creating the credential.  
  - Refresh |
  - AwsSv4  
  - Custom  
  - Jwt  
  - OAuth |
### Enum

<table>
<thead>
<tr>
<th>Enum</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>• AwsSv4_STS</td>
<td>AWS Signature Version 4 with Security Token Service.</td>
</tr>
<tr>
<td>• ClientCredentialsClientSecret</td>
<td>OAuth 2.0 Client Credentials client secret.</td>
</tr>
<tr>
<td>• ClientCredentialsJwtAssertion</td>
<td>OAuth 2.0 Client Credentials JSON Web Token assertion.</td>
</tr>
<tr>
<td>• JwtBearer</td>
<td>OAuth 2.0 JSON Web Token bearer flow.</td>
</tr>
<tr>
<td>• NoAuthentication</td>
<td>No authentication.</td>
</tr>
<tr>
<td>• RolesAnywhere</td>
<td>AWS Signature Version 4 with Identity and Access Management (IAM) Roles Anywhere.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>• Configured</td>
<td>Credential has all required credentials for at least one principal.</td>
</tr>
<tr>
<td>• NotConfigured</td>
<td>Credential isn’t configured.</td>
</tr>
<tr>
<td>• Unknown</td>
<td>Credential status can’t be determined because the authentication protocol is custom.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ConnectApi.CredentialPrincipalType</th>
<th>Type of credential principal.</th>
</tr>
</thead>
<tbody>
<tr>
<td>• AwsStsPrincipal</td>
<td></td>
</tr>
<tr>
<td>• NamedPrincipal</td>
<td></td>
</tr>
<tr>
<td>• PerUserPrincipal</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ConnectApi.DatacloudUserType</th>
<th>Type of user.</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Monthly</td>
<td>A user type that’s assigned monthly point limits for purchasing Data.com records. Only the assigned user can use monthly points. Points expire at the end of the month. Monthly is the default setting for DatacloudUserType.</td>
</tr>
<tr>
<td>• Listpool</td>
<td>A user type that allows users to draw from a pool of points to purchase Data.com records.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ConnectApi.DatacloudImportStatusTypeEnum</th>
<th>Status of the import.</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Success</td>
<td>Indicates that selected records were added to the org’s CRM.</td>
</tr>
<tr>
<td>• Duplicate</td>
<td>Marks a record that is already in the org’s CRM. The API determines whether an org allows the addition of duplicate records in its CRM.</td>
</tr>
<tr>
<td>• Error</td>
<td>Indicates that the selected records weren’t added to the org’s CRM.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ConnectApi.DigestPeriod</th>
<th>Time period that’s included in a Chatter email digest.</th>
</tr>
</thead>
<tbody>
<tr>
<td>• DailyDigest</td>
<td>The email includes up to the 50 latest posts from the previous day.</td>
</tr>
<tr>
<td>• WeeklyDigest</td>
<td>The email includes up to the 50 latest posts from the previous week.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ConnectApi.EmailMessageDirection</th>
<th>Direction of an email message on a case.</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Inbound</td>
<td>An inbound message (sent by a customer).</td>
</tr>
<tr>
<td>• Outbound</td>
<td>An outbound message (sent to a customer by a support agent).</td>
</tr>
<tr>
<td>Enum</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ConnectApi.EmailMessageStatus</td>
<td>Status of an email message on a case.</td>
</tr>
<tr>
<td>• DraftStatus</td>
<td></td>
</tr>
<tr>
<td>• ForwardedStatus</td>
<td></td>
</tr>
<tr>
<td>• NewStatus</td>
<td></td>
</tr>
<tr>
<td>• ReadStatus</td>
<td></td>
</tr>
<tr>
<td>• RepliedStatus</td>
<td></td>
</tr>
<tr>
<td>• SentStatus</td>
<td></td>
</tr>
<tr>
<td>ConnectApi.ExtensionInformationType</td>
<td>Information type of the extension.</td>
</tr>
<tr>
<td>• Lightning</td>
<td></td>
</tr>
<tr>
<td>ConnectApi.ExternalCredential.ParameterType</td>
<td>Parameter type of the external credential.</td>
</tr>
<tr>
<td>• AuthParameter</td>
<td></td>
</tr>
<tr>
<td>• AuthProvider</td>
<td></td>
</tr>
<tr>
<td>• AuthProviderUrl</td>
<td></td>
</tr>
<tr>
<td>• AuthProviderUrlQueryParameter</td>
<td></td>
</tr>
<tr>
<td>• JwtBodyClaim</td>
<td></td>
</tr>
<tr>
<td>• JwtHeaderClaim</td>
<td></td>
</tr>
<tr>
<td>• SigningCertificate</td>
<td></td>
</tr>
<tr>
<td>ConnectApi.ExternalCredential.PrincipalAccessType</td>
<td>Access type of the external credential principal.</td>
</tr>
<tr>
<td>• PermissionSet</td>
<td></td>
</tr>
<tr>
<td>• PermissionSetGroup</td>
<td></td>
</tr>
<tr>
<td>• Profile</td>
<td></td>
</tr>
<tr>
<td>ConnectApi.FeedCommentSortOrder</td>
<td>Order of comments.</td>
</tr>
<tr>
<td>• CreatedDateLatestAsc—Sorts by most recently created comments in ascending order.</td>
<td></td>
</tr>
<tr>
<td>• CreatedDateOldestAsc—Sorts by oldest comments in ascending order.</td>
<td></td>
</tr>
<tr>
<td>• Relevance—Sorts by most relevant content.</td>
<td></td>
</tr>
<tr>
<td>ConnectApi.FeedDensity</td>
<td>Density of the feed.</td>
</tr>
<tr>
<td>• AllUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations.</td>
<td></td>
</tr>
<tr>
<td>• FewerUpdates—Displays all updates from people and records the user follows and groups the user is a member of. Also displays custom recommendations, but hides some system-generated updates from records.</td>
<td></td>
</tr>
</tbody>
</table>
### Enum

**ConnectApi.FeedElementCapability Type**

- **Description**

  Capabilities of a feed element in API versions 31.0 and later. If a capability exists on a feed element, the capability is available, even if the value doesn’t exist or is `null`. If the capability doesn’t exist, it isn’t available.

  - **AssociatedActions**—The feed element includes information about actions associated with it.
  - **Approval**—The feed element includes information about an approval.
  - **Banner**—The body of the feed element has an icon and border.
  - **Bookmarks**—The context user can bookmark the feed element. Bookmarked feed elements are visible in the bookmarks feed.
  - **Bundle**—The feed element has a group of other feed elements that display as a bundle in the feed. The **bundle type** determines the additional data associated with the bundle.
  - **CallCollaboration**—The feed element has a recording comment.
  - **Canvas**—The feed element renders a canvas app.
  - **CaseComment**—The feed element has a case comment in the case feed.
  - **ChatterLikes**—The context user can like the feed element.
  - **Close**—The feed element can’t be edited, commented on, or deleted. If the feed element is a poll, it can’t be voted on.
  - **Comments**—The context user can add comments to the feed element.
  - **Content**—The feed element has a file.
  - **DashboardComponentSnapshot**—The feed element has a dashboard component snapshot.
  - **DirectMessage**—The feed element is a direct message.
  - **Edit**—Users who have permission can edit the feed element.
  - **EmailMessage**—The feed element has an email message from a case.
  - **EnhancedLink**—The feed element has a link that can contain supplemental information like an icon, a title, and a description.
  - **Extensions**—The feed element has one or more extension attachments.
  - **FeedEntityShare**—The feed element has a feed entity shared with it.
  - **Files**—The feed element has one or more file attachments.
  - **Interactions**—The feed element has information about user interactions.
  - **Link**—The feed element has a URL.
  - **MediaReferences**—The feed element has one or more media references.
  - **Moderation**—Users in an Experience Cloud site can flag the feed element for moderation.
  - **Mute**—The context user can mute the feed element.
  - **Origin**—A feed action created the feed element.
  - **Pin**—Users who have permission can pin the feed element.
  - **Poll**—The feed element has poll voting.
  - **QuestionAndAnswers**—The feed element has a question, and users can add answers to the feed element instead of comments. Users can also select the best answer.
  - **ReadBy**—The context user can mark the feed element as read.
<table>
<thead>
<tr>
<th>Enum</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConnectApi.FeedElement</td>
<td>Feed elements are the top-level objects that a feed contains. The feed element type describes the characteristics of that feed element.</td>
</tr>
<tr>
<td>• Bundle</td>
<td>A container of feed elements. A bundle also has a body made up of message segments that can always be gracefully degraded to text-only values.</td>
</tr>
<tr>
<td>• FeedItem</td>
<td>A feed item has a single parent and is scoped to one Experience Cloud site or across all Experience Cloud sites. A feed item can have capabilities such as bookmarks, canvas, content, comment, link, poll. Feed items have a body made up of message segments that can always be gracefully degraded to text-only values.</td>
</tr>
<tr>
<td>• Recommendation</td>
<td>A recommendation is a feed element with a recommendations capability. A recommendation suggests records to follow, groups to join, or applications that are helpful to the context user.</td>
</tr>
<tr>
<td>ConnectApi.FeedEntity</td>
<td>Status of the feed post or comment.</td>
</tr>
<tr>
<td>• Draft</td>
<td>The feed post isn’t published but is visible to the author and users with Modify All Data or View All Data permission. Comments can’t be drafts.</td>
</tr>
<tr>
<td>• PendingReview</td>
<td>The feed post or comment isn’t approved yet and therefore isn’t published or visible.</td>
</tr>
<tr>
<td>• Published</td>
<td>The feed post or comment is approved and visible.</td>
</tr>
<tr>
<td>ConnectApi.FeedFavorite</td>
<td>Origin of the feed favorite.</td>
</tr>
<tr>
<td>• ListView</td>
<td></td>
</tr>
<tr>
<td>• Search</td>
<td></td>
</tr>
<tr>
<td>• Topic</td>
<td></td>
</tr>
<tr>
<td>ConnectApi.FeedFilter</td>
<td>Filter value for a feed.</td>
</tr>
<tr>
<td>• AllQuestions</td>
<td>Feed elements that are questions.</td>
</tr>
<tr>
<td>• AuthoredBy</td>
<td>Feed elements authored by the user profile owner. This value is valid only for the UserProfile feed.</td>
</tr>
<tr>
<td>• CommunityScoped</td>
<td>Feed elements that are scoped to Experience Cloud sites. Currently, these feed elements have a User or a Group parent record. However, other parent record</td>
</tr>
<tr>
<td>Enum</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td></td>
<td>types could be scoped to sites in the future. Feed elements that are always visible in all sites are filtered out. This value is valid only for the UserProfile feed.</td>
</tr>
<tr>
<td></td>
<td>QuestionsWithCandidateAnswers—Feed elements that are questions that have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.</td>
</tr>
<tr>
<td></td>
<td>QuestionsWithCandidateAnswersReviewedPublished—Feed elements that are questions that have candidate answers that have been reviewed or published. This value is valid only for users with the Access Einstein-Generated Answers permission.</td>
</tr>
<tr>
<td></td>
<td>Read—Feed elements that are older than 30 days or are marked as read for the context user. Includes existing feed elements when the context user joined the group. This value is valid only for the Record feed of a group.</td>
</tr>
<tr>
<td></td>
<td>SolvedQuestions—Feed elements that are questions and that have a best answer.</td>
</tr>
<tr>
<td></td>
<td>UnansweredQuestions—Feed elements that are questions and that don’t have any answers.</td>
</tr>
<tr>
<td></td>
<td>UnansweredQuestionsWithCandidateAnswers—Feed elements that are questions that don’t have answers but have candidate answers associated with them. This value is valid only for users with the Access Einstein-Generated Answers permission.</td>
</tr>
<tr>
<td></td>
<td>Unread—Feed elements that are created in the past 30 days and aren’t marked as read for the context user. This value is valid only for the Record feed of a group.</td>
</tr>
<tr>
<td></td>
<td>UnsolvedQuestions—Feed elements that are questions and that don’t have a best answer.</td>
</tr>
<tr>
<td>ConnectApi.FeedItemAttachmentType</td>
<td>Attachment type for feed item output objects.</td>
</tr>
<tr>
<td></td>
<td>Approval—a feed item requiring approval.</td>
</tr>
<tr>
<td></td>
<td>BasicTemplate—a feed item with a generic rendering of an image, link, and title.</td>
</tr>
<tr>
<td></td>
<td>Canvas—a feed item that contains the metadata to render a link to a canvas app.</td>
</tr>
<tr>
<td></td>
<td>CaseComment—a feed item created from a comment to a case record.</td>
</tr>
<tr>
<td></td>
<td>CaseComment—a feed item created from a comment to a case record.</td>
</tr>
<tr>
<td></td>
<td>Content—a feed item with a file attached.</td>
</tr>
<tr>
<td></td>
<td>DashboardComponent—a feed item with a dashboard attached.</td>
</tr>
<tr>
<td></td>
<td>EmailMessage—an email attached to a case record in Case Feed.</td>
</tr>
<tr>
<td></td>
<td>Link—a feed item with a URL attached.</td>
</tr>
<tr>
<td></td>
<td>Poll—a feed item with a poll attached.</td>
</tr>
<tr>
<td></td>
<td>Question—a feed item with a question attached.</td>
</tr>
<tr>
<td></td>
<td>RecordSnapshot—the feed item attachment contains a view of a record at a single ConnectApi.FeedItemType.CreateRecordEvent.</td>
</tr>
<tr>
<td></td>
<td>TrackedChange—all changes to a record for a single ConnectApi.FeedItemType.TrackedChange event.</td>
</tr>
<tr>
<td>ConnectApi.FeedItemType</td>
<td>Type of feed item.</td>
</tr>
<tr>
<td></td>
<td>ActivityEvent—Feed item generated in Case Feed when an event or task associated with a parent record with a feed enabled is created or updated.</td>
</tr>
<tr>
<td>Enum</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>AdvancedTextPost</td>
<td>A feed item with advanced text formatting, such as a group announcement post.</td>
</tr>
<tr>
<td>ApprovalPost</td>
<td>Feed item with an approval capability. Approvers can act on the feed item parent.</td>
</tr>
<tr>
<td>AttachArticleEvent</td>
<td>Feed item generated when an article is attached to a case in Case Feed.</td>
</tr>
<tr>
<td>BasicTemplateFeedItem</td>
<td>Feed item with an enhanced link capability.</td>
</tr>
<tr>
<td>CallLogPost</td>
<td>Feed item generated when a call log is saved to a case in Case Feed.</td>
</tr>
<tr>
<td>CanvasPost</td>
<td>Feed item generated by a canvas app in the publisher or from Connect REST API or Connect in Apex. The post itself is a link to a canvas app.</td>
</tr>
<tr>
<td>CaseCommentPost</td>
<td>Feed item generated when a case comment is saved in Case Feed.</td>
</tr>
<tr>
<td>ChangeStatusPost</td>
<td>Feed item generated when the status of a case is changed in Case Feed.</td>
</tr>
<tr>
<td>ChatTranscriptionPost</td>
<td>Feed item generated in Case Feed when a Live Agent chat transcript is saved to a case.</td>
</tr>
<tr>
<td>CollaborationGroupCreated</td>
<td>Feed item generated when a new public group is created. Contains a link to the new group.</td>
</tr>
<tr>
<td>CollaborationGroupUnarchived</td>
<td>Deprecated. Feed item generated when an archived group is activated.</td>
</tr>
<tr>
<td>ContentPost</td>
<td>Feed item with a content capability.</td>
</tr>
<tr>
<td>CreateRecordEvent</td>
<td>Feed item that describes a record created in the publisher.</td>
</tr>
<tr>
<td>DashboardComponentAlert</td>
<td>Feed item with a dashboard alert.</td>
</tr>
<tr>
<td>DashboardComponentSnapshot</td>
<td>Feed item with a dashboard component snapshot capability.</td>
</tr>
<tr>
<td>EmailMessageEvent</td>
<td>Feed item generated when an email is sent from a case in Case Feed.</td>
</tr>
<tr>
<td>FacebookPost</td>
<td>Deprecated. Feed item generated when a Facebook post is created from a case in Case Feed.</td>
</tr>
<tr>
<td>LinkPost</td>
<td>Feed item with a link capability.</td>
</tr>
<tr>
<td>MilestoneEvent</td>
<td>Feed item generated when a case milestone is either completed or reaches a violation status. Contains a link to the case milestone.</td>
</tr>
<tr>
<td>PollPost</td>
<td>Feed item with a poll capability. Viewers of the feed item are allowed to vote on the options in the poll.</td>
</tr>
<tr>
<td>ProfileSkillPost</td>
<td>Feed item generated when a skill is added to a user’s profile.</td>
</tr>
<tr>
<td>QuestionPost</td>
<td>Feed item generated when a question is asked.</td>
</tr>
<tr>
<td></td>
<td>As of API version 33.0, a feed item of this type can have a content capability and a link capability.</td>
</tr>
<tr>
<td>ReplyPost</td>
<td>Feed item generated by a Chatter Answers reply.</td>
</tr>
<tr>
<td>RypplePost</td>
<td>Feed item generated when a user posts thanks.</td>
</tr>
<tr>
<td>SocialPost</td>
<td>Feed item generated when a social post is created from a case in Case Feed.</td>
</tr>
<tr>
<td>Enum</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>ConnectApi.FeedItemVisibilityType</td>
<td>Type of users who can see a feed item.</td>
</tr>
<tr>
<td>AllUsers</td>
<td>Visibility is not limited to internal users.</td>
</tr>
<tr>
<td>InternalUsers</td>
<td>Visibility is limited to internal users.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ConnectApi.FeedSortOrder</th>
<th>Order of feed items in the feed.</th>
</tr>
</thead>
<tbody>
<tr>
<td>CreatedDateAsc</td>
<td>Sorts by oldest creation date. This sort order is available only for DirectMessageModeration, Draft, Moderation, and PendingReview feeds.</td>
</tr>
<tr>
<td>CreatedDateDesc</td>
<td>Sorts by most recent creation date.</td>
</tr>
<tr>
<td>LastModifiedDateDesc</td>
<td>Sorts by most recent activity.</td>
</tr>
<tr>
<td>MostViewed</td>
<td>Sorts by most viewed content. This sort order is available only for Home feeds when the ConnectApi.FeedFilter is UnansweredQuestions.</td>
</tr>
<tr>
<td>Relevance</td>
<td>Sorts by most relevant content. This sort order is available only for Company, Home, and Topics feeds.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ConnectApi.FeedType</th>
<th>Type of feed.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bookmarks</td>
<td>Contains all feed items saved as bookmarks by the context user.</td>
</tr>
<tr>
<td>Company</td>
<td>Contains all feed items except feed items of type TrackedChange. To see the feed item, the user must have sharing access to its parent.</td>
</tr>
<tr>
<td>DirectMessageModeration</td>
<td>Contains all direct messages that are flagged for moderation. The Direct Message Moderation feed is available only to users with Moderate Experiences Chatter Messages permissions.</td>
</tr>
<tr>
<td>DirectMessages</td>
<td>Contains all feed items of the context user’s direct messages.</td>
</tr>
<tr>
<td>Draft</td>
<td>Contains all the feed items that the context user drafted.</td>
</tr>
<tr>
<td>Files</td>
<td>Contains all feed items that contain files posted by people or groups that the context user follows.</td>
</tr>
<tr>
<td>Filter</td>
<td>Contains the news feed filtered to contain feed items whose parent is a specified object type.</td>
</tr>
<tr>
<td>Groups</td>
<td>Contains all feed items from all groups the context user either owns or is a member of.</td>
</tr>
<tr>
<td>Home</td>
<td>Contains all feed items associated with any managed topic in an Experience Cloud site.</td>
</tr>
<tr>
<td>Landing</td>
<td>Contains all feed items that best drive user engagement when the feed is requested. Allows clients to avoid an empty feed when there aren’t many personalized feed items.</td>
</tr>
<tr>
<td>Enum</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Moderation</td>
<td>Contains all feed items that are flagged for moderation, except direct messages. The moderation feed is available only to users with Moderate Experiences Feeds permissions.</td>
</tr>
<tr>
<td>Mute</td>
<td>Contains all feed items that the context user muted.</td>
</tr>
<tr>
<td>News</td>
<td>Contains all updates for people the context user follows, groups the user is a member of, and files and records the user is following. Contains all updates for records whose parent is the context user.</td>
</tr>
<tr>
<td>PendingReview</td>
<td>Contains all feed items and comments that are pending review.</td>
</tr>
<tr>
<td>People</td>
<td>Contains all feed items posted by all people the context user follows.</td>
</tr>
<tr>
<td>Record</td>
<td>Contains all feed items whose parent is a specified record, which could be a group, user, object, file, or any other standard or custom object. When the record is a group, the feed also contains feed items that mention the group. When the record is a user, the feed contains only feed items on that user. You can get another user’s record feed.</td>
</tr>
<tr>
<td>Streams</td>
<td>Contains all feed items for any combination of up to 25 feed-enabled entities that the context user subscribes to in a stream. Examples of feed-enabled entities include people, groups, and records.</td>
</tr>
<tr>
<td>To</td>
<td>Contains all feed items with mentions of the context user. Contains feed items the context user commented on and feed items created by the context user that are commented on.</td>
</tr>
<tr>
<td>Topics</td>
<td>Contains all feed items that include the specified topic.</td>
</tr>
<tr>
<td>UserProfile</td>
<td>Contains feed items created when a user changes records that can be tracked in a feed. Contains feed items whose parent is the user and feed items that @mention the user. This feed is different than the news feed, which returns more feed items, including group updates. You can get another user’s user profile feed.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ConnectApi.FieldChange ValueType</th>
<th>Value type of a field change.</th>
</tr>
</thead>
<tbody>
<tr>
<td>NewValue</td>
<td>A new value</td>
</tr>
<tr>
<td>OldValue</td>
<td>An old value</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ConnectApi.FilePreviewFormat</th>
<th>Format of the file preview.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jpg</td>
<td>Preview format is JPG.</td>
</tr>
<tr>
<td>Pdf</td>
<td>Preview format is PDF.</td>
</tr>
<tr>
<td>Svg</td>
<td>Preview format is compressed SVG.</td>
</tr>
<tr>
<td>Thumbnail</td>
<td>Preview format is 240 x 180 PNG.</td>
</tr>
<tr>
<td>ThumbnailBig</td>
<td>Preview format is 720 x 480 PNG.</td>
</tr>
<tr>
<td>ThumbnailTiny</td>
<td>Preview format is 120 x 90 PNG.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ConnectApi.FilePreviewStatus</th>
<th>Availability status of the file preview.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Available</td>
<td>Preview is available.</td>
</tr>
<tr>
<td>InProgress</td>
<td>Preview is being processed.</td>
</tr>
<tr>
<td>NotAvailable</td>
<td>Preview is unavailable.</td>
</tr>
<tr>
<td>NotScheduled</td>
<td>Generation of the preview isn’t scheduled yet.</td>
</tr>
<tr>
<td>Enum</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ConnectApi.FilePublishStatus</td>
<td>Publish status of the file.</td>
</tr>
<tr>
<td></td>
<td>• PendingAccess—File is pending publishing.</td>
</tr>
<tr>
<td></td>
<td>• PrivateAccess—File is private.</td>
</tr>
<tr>
<td></td>
<td>• PublicAccess—File is public.</td>
</tr>
<tr>
<td>ConnectApi.FileSharingOption</td>
<td>Sharing option of the file.</td>
</tr>
<tr>
<td></td>
<td>• Allowed—Resharing of the file is allowed.</td>
</tr>
<tr>
<td></td>
<td>• Restricted—Resharing of the file is restricted.</td>
</tr>
<tr>
<td></td>
<td>• None—File is visible to anyone with record access.</td>
</tr>
<tr>
<td></td>
<td>• PrivateOnRecords—File is private on records.</td>
</tr>
<tr>
<td>ConnectApi.FileSharingType</td>
<td>Sharing role of the file.</td>
</tr>
<tr>
<td></td>
<td>• Admin—Owner permission, but doesn’t own the file.</td>
</tr>
<tr>
<td></td>
<td>• Collaborator—Viewer permission, and can edit, change permissions, and upload a new version of a file.</td>
</tr>
<tr>
<td></td>
<td>• Owner—Collaborator permission, and can make a file private, and delete a file.</td>
</tr>
<tr>
<td></td>
<td>• Viewer—Can view, download, and share a file.</td>
</tr>
<tr>
<td></td>
<td>• WorkspaceManaged—Permission controlled by the library.</td>
</tr>
<tr>
<td>ConnectApi.FolderItemType</td>
<td>Type of item in a folder.</td>
</tr>
<tr>
<td></td>
<td>• file</td>
</tr>
<tr>
<td></td>
<td>• folder</td>
</tr>
<tr>
<td>ConnectApi.FormFieldType</td>
<td>Type of marketing integration form field.</td>
</tr>
<tr>
<td></td>
<td>• Boolean</td>
</tr>
<tr>
<td></td>
<td>• Date</td>
</tr>
<tr>
<td></td>
<td>• EmailAddress</td>
</tr>
<tr>
<td></td>
<td>• Number</td>
</tr>
<tr>
<td></td>
<td>• Text</td>
</tr>
<tr>
<td>ConnectApi.FormulaFilterType</td>
<td>Formula filter type for the personalization audience.</td>
</tr>
<tr>
<td></td>
<td>• AllCriteriaMatch—All audience criteria are true (AND operation).</td>
</tr>
<tr>
<td></td>
<td>• AnyCriterionMatches—Any audience criterion is true (OR operation).</td>
</tr>
<tr>
<td></td>
<td>• CustomLogicMatches—Audience criteria match the custom formula (for example, (1 AND 2) OR 3).</td>
</tr>
<tr>
<td>ConnectApi.GroupArchiveStatus</td>
<td>Archive status of groups.</td>
</tr>
<tr>
<td></td>
<td>• All—All groups, including groups that are archived and groups that aren’t archived.</td>
</tr>
<tr>
<td></td>
<td>• Archived—Groups that are archived.</td>
</tr>
<tr>
<td>Enum</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ConnectApi.GroupEmailFrequency</td>
<td>Frequency with which a user receives email.</td>
</tr>
<tr>
<td></td>
<td>• EachPost</td>
</tr>
<tr>
<td></td>
<td>• DailyDigest</td>
</tr>
<tr>
<td></td>
<td>• WeeklyDigest</td>
</tr>
<tr>
<td></td>
<td>• Never</td>
</tr>
<tr>
<td></td>
<td>• UseDefault</td>
</tr>
<tr>
<td>ConnectApi.GroupMembershipType</td>
<td>Type of membership the user has with the group.</td>
</tr>
<tr>
<td></td>
<td>• GroupOwner</td>
</tr>
<tr>
<td></td>
<td>• GroupManager</td>
</tr>
<tr>
<td></td>
<td>• NotAMember</td>
</tr>
<tr>
<td></td>
<td>• NotAMemberPrivateRequested</td>
</tr>
<tr>
<td></td>
<td>• StandardMember</td>
</tr>
<tr>
<td>ConnectApi.GroupMembershipRequestStatus</td>
<td>Status of a request to join a private group.</td>
</tr>
<tr>
<td></td>
<td>• Accepted</td>
</tr>
<tr>
<td></td>
<td>• Declined</td>
</tr>
<tr>
<td></td>
<td>• Pending</td>
</tr>
<tr>
<td>ConnectApi.GroupViralInvitationsStatus</td>
<td>Status of an invitation to join a group.</td>
</tr>
<tr>
<td></td>
<td>• ActedUponUser—The user was added to the group. An email was sent asking the user to visit the group.</td>
</tr>
<tr>
<td></td>
<td>• Invited—An email was sent asking the user to sign up for the org.</td>
</tr>
<tr>
<td></td>
<td>• MaxedOutUsers—The group has the maximum allowed members.</td>
</tr>
<tr>
<td></td>
<td>• MultipleError—The user wasn’t invited due to multiple errors.</td>
</tr>
<tr>
<td></td>
<td>• NoActionNeededUser—The user is already a member of the group.</td>
</tr>
<tr>
<td></td>
<td>• NotVisibleToExternalInviter—The user is not accessible to the user sending the invitation.</td>
</tr>
<tr>
<td></td>
<td>• Unhandled—The user couldn’t be added to the group for an unknown reason.</td>
</tr>
<tr>
<td>ConnectApi.GroupVisibilityType</td>
<td>Group visibility type.</td>
</tr>
<tr>
<td></td>
<td>• PrivateAccess—Only members of the group can see posts to this group.</td>
</tr>
<tr>
<td></td>
<td>• PublicAccess—All users within the Experience Cloud site can see posts to this group.</td>
</tr>
<tr>
<td></td>
<td>• Unlisted—Reserved for future use.</td>
</tr>
<tr>
<td>ConnectApi.HttpRequestMethod</td>
<td>HTTP method.</td>
</tr>
<tr>
<td></td>
<td>• HttpDelete—Returns HTTP 204 on success. Response body or output class is empty.</td>
</tr>
<tr>
<td></td>
<td>• HttpGet—Returns HTTP 200 on success.</td>
</tr>
<tr>
<td></td>
<td>• HttpHead—Returns HTTP 200 on success. Response body or output class is empty.</td>
</tr>
<tr>
<td>Enum</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>HttpPatch</td>
<td>Returns HTTP 200 on success or HTTP 204 if the response body or output class is empty.</td>
</tr>
<tr>
<td>HttpPost</td>
<td>Returns HTTP 201 on success or HTTP 204 if the response body or output class is empty. Exceptions are the batch posting resources and methods, which return HTTP 200 on success.</td>
</tr>
<tr>
<td>HttpPut</td>
<td>Return HTTP 200 on success or HTTP 204 if the response body or output class is empty.</td>
</tr>
<tr>
<td>Source of the link metadata</td>
<td>ConnectApi. LinkMetadataSource</td>
</tr>
<tr>
<td>None</td>
<td>Link metadata wasn’t retrieved.</td>
</tr>
<tr>
<td>Sfdc</td>
<td>Salesforce is the source.</td>
</tr>
<tr>
<td>ConnectApi. LinkMetadataType</td>
<td>Type of link that the metadata represents.</td>
</tr>
<tr>
<td>Error</td>
<td>Link metadata couldn’t be retrieved.</td>
</tr>
<tr>
<td>Link</td>
<td>Represents a link.</td>
</tr>
<tr>
<td>None</td>
<td>Link metadata wasn’t retrieved because the link isn’t an allowed domain.</td>
</tr>
<tr>
<td>Photo</td>
<td>Represents a photo.</td>
</tr>
<tr>
<td>Rich</td>
<td>Represents rich content, typically HTML content.</td>
</tr>
<tr>
<td>Unknown</td>
<td>Link metadata was retrieved, but the type is unknown.</td>
</tr>
<tr>
<td>Video</td>
<td>Represents a video.</td>
</tr>
<tr>
<td>ConnectApi. MaintenanceType</td>
<td>Type of maintenance.</td>
</tr>
<tr>
<td>Downtime</td>
<td>Downtime maintenance.</td>
</tr>
<tr>
<td>GenerallyAvailable</td>
<td>Maintenance with generally available mode.</td>
</tr>
<tr>
<td>MaintenanceWithDowntime</td>
<td>Scheduled maintenance with downtime.</td>
</tr>
<tr>
<td>ReadOnly</td>
<td>Maintenance with read-only mode.</td>
</tr>
<tr>
<td>ConnectApi. ManagedContentChannelType</td>
<td>Type of managed content channel.</td>
</tr>
<tr>
<td>CloudToCloud</td>
<td>Cloud-to-Cloud integrated channel.</td>
</tr>
<tr>
<td>Community</td>
<td>Experience Cloud site channel.</td>
</tr>
<tr>
<td>ConnectedApp</td>
<td>Channel served by a connected app.</td>
</tr>
<tr>
<td>PublicUnauthenticated</td>
<td>Public channel. All published content is publicly available.</td>
</tr>
<tr>
<td>UserPermission</td>
<td>Channel backed by a system permission. All published content is available only to users with the permission.</td>
</tr>
<tr>
<td>ConnectApi. ManagedContentMediaType</td>
<td>Type of managed content media.</td>
</tr>
<tr>
<td>Document</td>
<td></td>
</tr>
<tr>
<td>Image</td>
<td></td>
</tr>
<tr>
<td>ConnectApi. ManagedContentNodeType</td>
<td>Type of managed content node.</td>
</tr>
<tr>
<td>Date</td>
<td></td>
</tr>
<tr>
<td>Enum</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>• DateTime</td>
<td>Type of managed topic.</td>
</tr>
<tr>
<td>• Media</td>
<td>• Content—Topics that are associated with native content.</td>
</tr>
<tr>
<td>• MediaSource</td>
<td>• Featured—Topics that are featured, for example, on the Experience Cloud site home page, but don’t provide overall navigation.</td>
</tr>
<tr>
<td>• MultilineText</td>
<td>• Navigational—Topics that display in a navigational menu in the Experience Cloud site.</td>
</tr>
<tr>
<td>• NameField</td>
<td>• RichText</td>
</tr>
<tr>
<td>• Text</td>
<td>• Url</td>
</tr>
<tr>
<td>• Url</td>
<td><strong>ConnectApi.ManagedTopicType</strong> Type of managed topic.</td>
</tr>
<tr>
<td>• ConnectApi.MarkupType</td>
<td>• Bold—Bold tag.</td>
</tr>
<tr>
<td></td>
<td>• Code—Code tag.</td>
</tr>
<tr>
<td></td>
<td>• Hyperlink—Hyperlink anchor tag.</td>
</tr>
<tr>
<td></td>
<td>• Italic—Italic tag.</td>
</tr>
<tr>
<td></td>
<td>• ListItem—List item tag.</td>
</tr>
<tr>
<td></td>
<td>• OrderedList—Ordered list tag.</td>
</tr>
<tr>
<td></td>
<td>• Paragraph—Paragraph tag.</td>
</tr>
<tr>
<td></td>
<td>• Strikethrough—Strikethrough tag.</td>
</tr>
<tr>
<td></td>
<td>• Underline—Underline tag.</td>
</tr>
<tr>
<td></td>
<td>• UnorderedList—Unordered list tag.</td>
</tr>
<tr>
<td></td>
<td><strong>ConnectApi.MentionCompletionType</strong> Type of mention completion.</td>
</tr>
<tr>
<td></td>
<td>• All—All mention completions, regardless of the type of record to which the mention refers.</td>
</tr>
<tr>
<td></td>
<td>• Group—Mention completions for groups.</td>
</tr>
<tr>
<td></td>
<td>• User—Mention completions for users.</td>
</tr>
<tr>
<td></td>
<td><strong>ConnectApi.MentionValidationStatus</strong> Type of validation error for a proposed mention, if any.</td>
</tr>
<tr>
<td></td>
<td>• Disallowed—The proposed mention is invalid and is rejected because the context user is trying to mention something that is not allowed. For example, a user who is not a member of a private group is trying to mention the private group.</td>
</tr>
<tr>
<td></td>
<td>• Inaccessible—The proposed mention is allowed, but the user or record being mentioned isn’t notified. They don’t have access to the parent record that’s being discussed.</td>
</tr>
<tr>
<td></td>
<td>• Ok—There is no validation error for this proposed mention.</td>
</tr>
<tr>
<td>Enum</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>ConnectApi. MessageSegmentType</td>
<td>Type of message segment, such as text, link, field change name, or field change value.</td>
</tr>
<tr>
<td>EntityLink</td>
<td></td>
</tr>
<tr>
<td>FieldChange</td>
<td></td>
</tr>
<tr>
<td>FieldChangeName</td>
<td></td>
</tr>
<tr>
<td>FieldChangeValue</td>
<td></td>
</tr>
<tr>
<td>Hashtag</td>
<td></td>
</tr>
<tr>
<td>InlineImage</td>
<td></td>
</tr>
<tr>
<td>Link</td>
<td></td>
</tr>
<tr>
<td>MarkupBegin</td>
<td></td>
</tr>
<tr>
<td>MarkupEnd</td>
<td></td>
</tr>
<tr>
<td>Mention</td>
<td></td>
</tr>
<tr>
<td>MoreChanges</td>
<td></td>
</tr>
<tr>
<td>ResourceLink</td>
<td></td>
</tr>
<tr>
<td>Text</td>
<td></td>
</tr>
</tbody>
</table>

| ConnectApi. NamedCredential ParameterType | Type of named credential parameter. |
| AllowedManagedPackageNamespaces | |
| ClientCertificate | |

| ConnectApi. NamedCredentialType | Type of named credential. |
| PrivateEndpoint | |
| SecuredEndpoint | |

| ConnectApi. NavigationMenuItem ActionType | Event, URL type, or modal navigation menu item. |
| Event—Event-based navigation. | |
| Note: Event is internal only and can’t be used in custom components. | |
| ExternalLink—URL outside of your Experience Cloud site. | |
| InternalLink—Relative URL inside your Experience Cloud site. | |
| Modal—Modal, such as Account Switcher. | |

| ConnectApi. NavigationMenuItem OpenTarget | Target for the navigation menu item. |
| CurrentWindow—Navigation menu item opens in the current window. | |
| NewWindow—Navigation menu item opens in a new window. | |

<p>| ConnectApi. NavigationMenuItemType | Type of navigation menu item. |
| DataSourceDriven—Menu items dynamically added from a data source. | |
| Event—Event, such as logging in, logging out, or switching accounts. | |
| ExternalLink—URL outside of your site. | |
| GlobalAction—Lets users create records that aren’t related to other records. | |</p>
<table>
<thead>
<tr>
<th>Enum</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>InternalLink</td>
<td>Relative URL inside your site.</td>
</tr>
<tr>
<td>MenuLabel</td>
<td>Menu label.</td>
</tr>
<tr>
<td>Modal</td>
<td>Modal, such as Account Switcher.</td>
</tr>
<tr>
<td>NavigationalTopic</td>
<td>Dropdown list with links to the navigational topics in your site.</td>
</tr>
<tr>
<td>SalesforceObject</td>
<td>Objects such as accounts, cases, contacts, and custom objects.</td>
</tr>
<tr>
<td>SystemLink</td>
<td>System link, such as a link to Builder, Workspaces, or Setup.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Enum</th>
<th>Type of action.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flow</td>
<td>Automated process tool with multiple subtypes.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Enum</th>
<th>Type of recommended flow.</th>
</tr>
</thead>
<tbody>
<tr>
<td>AutoLaunchedFlow</td>
<td>Autolaunched flow that runs in the background.</td>
</tr>
<tr>
<td>Flow</td>
<td>Screen flow that accepts user inputs.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Enum</th>
<th>Type of target.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recommendation</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Enum</th>
<th>Operation to carry out on the file.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add</td>
<td>Adds the file to the feed element.</td>
</tr>
<tr>
<td>Remove</td>
<td>Removes the file from the feed element.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Enum</th>
<th>Status of the orchestration instance.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Completed</td>
<td></td>
</tr>
<tr>
<td>InProgress</td>
<td></td>
</tr>
<tr>
<td>NotStarted</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Enum</th>
<th>Type of orchestration step.</th>
</tr>
</thead>
<tbody>
<tr>
<td>AsynchronousBackgroundStep</td>
<td></td>
</tr>
<tr>
<td>BackgroundStep</td>
<td></td>
</tr>
<tr>
<td>InteractiveStep</td>
<td></td>
</tr>
<tr>
<td>ManagedContentRoleInteractiveStep</td>
<td></td>
</tr>
<tr>
<td>ManagedContentVariantAutoPublishBackgroundStep</td>
<td></td>
</tr>
<tr>
<td>ManagedContentVariantAutoUnpublishBackgroundStep</td>
<td></td>
</tr>
<tr>
<td>ManagedContentVariantSetLockBackgroundStep</td>
<td></td>
</tr>
<tr>
<td>ManagedContentVariantSetReadyBackgroundStep</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Enum</th>
<th>Status of the orchestration work item.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assigned</td>
<td></td>
</tr>
<tr>
<td>Completed</td>
<td></td>
</tr>
<tr>
<td>Enum</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>ConnectApi.OrderDeliveryGroupSummarySort</td>
<td>Sort order for order delivery group summaries.</td>
</tr>
<tr>
<td></td>
<td>• IdAsc—Sorts by ID in ascending alphanumeric order (A–Z, 0–9).</td>
</tr>
<tr>
<td></td>
<td>• IdDesc—Sorts by ID in descending alphanumeric order (Z–A, 9–0).</td>
</tr>
<tr>
<td>ConnectApi.OrderItemSummarySort</td>
<td>Sort order for order item summaries.</td>
</tr>
<tr>
<td></td>
<td>• IdAsc—Sorts by ID in ascending alphanumeric order (A–Z, 0–9).</td>
</tr>
<tr>
<td></td>
<td>• IdDesc—Sorts by ID in descending alphanumeric order (Z–A, 9–0).</td>
</tr>
<tr>
<td>ConnectApi.OrderShipmentItemSort</td>
<td>Sort order for order shipment items.</td>
</tr>
<tr>
<td></td>
<td>• IdAsc—Sorts by ID in ascending alphanumeric order (A–Z, 0–9).</td>
</tr>
<tr>
<td></td>
<td>• IdDesc—Sorts by ID in descending alphanumeric order (Z–A, 9–0).</td>
</tr>
<tr>
<td>ConnectApi.OrderShipmentSort</td>
<td>Sort order for order shipments.</td>
</tr>
<tr>
<td></td>
<td>• ExpectedDeliveryDateAsc—Sorts by the oldest expected delivery date.</td>
</tr>
<tr>
<td></td>
<td>• ExpectedDeliveryDateDesc—Sorts by the most recent expected delivery date.</td>
</tr>
<tr>
<td></td>
<td>• ShipmentNumberAsc—Sorts by shipment number in ascending order (0–9).</td>
</tr>
<tr>
<td></td>
<td>• ShipmentNumberDesc—Sorts by shipment number in descending order (9–0).</td>
</tr>
<tr>
<td>ConnectApi.OrderSummaryAdjustmentAggregatesStatus</td>
<td>Order summary adjustment aggregate job status.</td>
</tr>
<tr>
<td></td>
<td>• Failed—The adjustment aggregate data job for the order summary failed.</td>
</tr>
<tr>
<td></td>
<td>• InProgress—The adjustment aggregate data job for the order summary is in progress.</td>
</tr>
<tr>
<td></td>
<td>• NotInitiated—The adjustment aggregate data job for the order summary is not initiated.</td>
</tr>
<tr>
<td></td>
<td>• Submitted—The adjustment aggregate data job for the order summary is submitted.</td>
</tr>
<tr>
<td>ConnectApi.OrderSummaryAdjustmentTargetType</td>
<td>Type of price adjustment in promotions.</td>
</tr>
<tr>
<td></td>
<td>• SplitLine—Price adjustment on an order item.</td>
</tr>
<tr>
<td></td>
<td>• Header—Price adjustment on the entire order.</td>
</tr>
<tr>
<td>ConnectApi.OrderSummarySortOrder</td>
<td>Sort order for order summaries.</td>
</tr>
<tr>
<td></td>
<td>• CreatedDateAsc—Sorts by the oldest created date.</td>
</tr>
<tr>
<td></td>
<td>• CreatedDateDesc—Sorts by the most recent created date.</td>
</tr>
<tr>
<td></td>
<td>• OrderedDateAsc—Sorts by the oldest ordered date.</td>
</tr>
<tr>
<td></td>
<td>• OrderedDateDesc—Sorts by the most recent ordered date.</td>
</tr>
<tr>
<td>ConnectApi.PeriodType</td>
<td>Time period used for forecasting.</td>
</tr>
<tr>
<td></td>
<td>• Month</td>
</tr>
<tr>
<td></td>
<td>• Quarter</td>
</tr>
<tr>
<td></td>
<td>• Week</td>
</tr>
<tr>
<td></td>
<td>• Year</td>
</tr>
<tr>
<td>Enum</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| ConnectApi. PlatformActionGroupCategory | Location of an action link group on an associated feed element.  
  - Primary—The action link group is displayed in the body of the feed element.  
  - Overflow—The action link group is displayed in the overflow menu of the feed element. |
  - FailedStatus—The action link execution failed.  
  - NewStatus—The action link is ready to be executed. Available for Download and Ui action links only.  
  - PendingStatus—The action link is executing. Choosing this value triggers the API call for Api and ApiAsync action links.  
  - SuccessfulStatus—The action link executed successfully. |
| ConnectApi. PlatformActionType | Type of platform action.  
  - ActionLink—An indicator on a feed element that targets an API, a web page, or a file, represented by a button in the Salesforce UI.  
  - CustomButton—When clicked, opens a URL or a Visualforce page in a window or executes JavaScript.  
  - ProductivityAction—Productivity actions are predefined and attached to a limited set of objects. Productivity actions include Send Email, Call, Map, View Website, and Read News. Except for the Call action, you can’t edit productivity actions.  
  - QuickAction—A global or object-specific action.  
  - StandardButton—A predefined Salesforce button such as New, Edit, or Delete. |
| ConnectApi. PriceAdjustmentTierType | Type of price adjustment for the tier.  
  - AmountBasedAdjustment—Price is adjusted by a specified amount.  
  - PercentageBasedAdjustment—Price is adjusted by a specified percentage. |
| ConnectApi. PricingTermUnit | Unit of time used to define a pricing term.  
  - Months—Product is priced on a monthly basis.  
  - Annual—Product is priced on an annual basis. |
| ConnectApi.ProductClass | Class of product.  
  - Simple  
  - Variation  
  - VariationParent |
| ConnectApi. ProductMediaType | Type of product media.  
  - Document  
  - Image  
  - Video |
### ConnectApi Enums

<table>
<thead>
<tr>
<th>Enum</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ProductMediaUsageType</strong></td>
<td>Usage type of a product media item within a media group.</td>
</tr>
<tr>
<td></td>
<td>• Attachment—Product media group with product documents as attachments.</td>
</tr>
<tr>
<td></td>
<td>• Banner—Product category media group with banner images of the product.</td>
</tr>
<tr>
<td></td>
<td>• Listing—Product media group with listing images of the product.</td>
</tr>
<tr>
<td></td>
<td>• Standard—Product media group with standard images and videos of the product.</td>
</tr>
<tr>
<td></td>
<td>• Tile—Product category media group with tile images of the product.</td>
</tr>
<tr>
<td><strong>PublishSchedule</strong></td>
<td>Publish refresh schedule.</td>
</tr>
<tr>
<td></td>
<td>• One—Refreshes every hour. Used to rapidly publish UI and DBT-based segments.</td>
</tr>
<tr>
<td></td>
<td>• Four—Refreshes every four hours. Used to rapidly publish UI and DBT-based segments.</td>
</tr>
<tr>
<td></td>
<td>• Twelve—Refreshes every twelve hours.</td>
</tr>
<tr>
<td></td>
<td>• TwentyFour—Refreshes every twenty-four hours.</td>
</tr>
<tr>
<td><strong>PublishStatus</strong></td>
<td>Publish status of a personalization audience, target, or navigation menu item.</td>
</tr>
<tr>
<td></td>
<td>• Draft</td>
</tr>
<tr>
<td></td>
<td>• Live</td>
</tr>
<tr>
<td><strong>RecommendationActionType</strong></td>
<td>Action to take on a recommendation.</td>
</tr>
<tr>
<td></td>
<td>• follow—Follow a file, record, topic, or user.</td>
</tr>
<tr>
<td></td>
<td>• join—Join a group.</td>
</tr>
<tr>
<td></td>
<td>• view—View a file, group, article, record, user, custom, or static recommendation.</td>
</tr>
<tr>
<td><strong>RecommendationAudienceCriteriaType</strong></td>
<td>Custom recommendation audience criteria type.</td>
</tr>
<tr>
<td></td>
<td>• CustomList—A custom list of users makes up the audience.</td>
</tr>
<tr>
<td></td>
<td>• MaxDaysInCommunity—New members make up the audience.</td>
</tr>
<tr>
<td><strong>RecommendationAudienceMemberOperationType</strong></td>
<td>Operation to carry out on the custom recommendation audience members.</td>
</tr>
<tr>
<td></td>
<td>• Add—Adds specified members to the audience.</td>
</tr>
<tr>
<td></td>
<td>• Remove—Removes specified members from the audience.</td>
</tr>
<tr>
<td><strong>RecommendationChannel</strong></td>
<td>A way to tie custom recommendations together. For example, display recommendations in specific places in the UI or show recommendations based on time of day or geographic locations.</td>
</tr>
<tr>
<td></td>
<td>• CustomChannel1—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels. For example, community managers can use Experience Builder to determine where recommendations appear.</td>
</tr>
<tr>
<td></td>
<td>• CustomChannel2—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.</td>
</tr>
<tr>
<td></td>
<td>• CustomChannel3—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.</td>
</tr>
<tr>
<td></td>
<td>• CustomChannel4—Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.</td>
</tr>
</tbody>
</table>
**Enumerations**

<table>
<thead>
<tr>
<th>Enum</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CustomChannel5</td>
<td>Custom recommendation channel. Not used by default. Work with your community manager to define custom channels.</td>
</tr>
<tr>
<td>DefaultChannel</td>
<td>Default recommendation channel. Recommendations appear by default on the Home and Question Detail pages of Customer Service and Partner Central Experience Builder templates. They also appear in the feed in the Salesforce mobile web and anywhere community managers add recommendations using Experience Builder.</td>
</tr>
</tbody>
</table>

**Reason for a Chatter recommendation.**

- ArticleHasRelatedContent—Articles with related content to a context article.
- ArticleViewedTogether—Articles often viewed together with the article that the context user just viewed.
- ArticleViewedTogetherWithViewers—Articles often viewed together with other records that the context user views.
- Custom—Custom recommendations.
- FilePopular—Files with many followers or views.
- FileViewedTogether—Files often viewed at the same time as other files that the context user views.
- FollowedTogetherWithFollowees—Users often followed together with other records that the context user follows.
- GroupMembersFollowed—Groups with members that the context user follows.
- GroupNew—Recently created groups.
- GroupPopular—Groups with many active members.
- ItemViewedTogether—Records often viewed at the same time as other records that the context user views.
- PopularApp—Applications that are popular.
- RecordOwned—Records that the context user owns.
- RecordParentOfFollowed—Parent records of records that the context user follows.
- RecordViewed—Records that the context user recently viewed.
- TopicFollowedTogether—Topics often followed together with the record that the context user just followed.
- TopicFollowedTogetherWithFollowees—Topics often followed together with other records that the context user follows.
- TopicPopularFollowed—Topics with many followers.
- TopicPopularLiked—Topics on posts that have many likes.
- UserDirectReport—Users who report to the context user.
- UserFollowedTogether—Users often followed together with the record that the context user followed.
- UserFollowsSameUsers—Users who follow the same users as the context user.
- UserManager—The context user’s manager.
- UserNew—Recently created users.
- UserPeer—Users who report to the same manager as the context user.
<table>
<thead>
<tr>
<th>Enum</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UserPopular—Users with many followers.</td>
<td></td>
</tr>
<tr>
<td>UserViewingSameRecords—Users who view the same records as the context user.</td>
<td></td>
</tr>
<tr>
<td>ConnectApi. RecommendationReactionType</td>
<td>Type of reaction to a recommendation.</td>
</tr>
<tr>
<td>Accepted</td>
<td></td>
</tr>
<tr>
<td>Rejected</td>
<td></td>
</tr>
<tr>
<td>ConnectApi. RecommendationType</td>
<td>Type of record being recommended.</td>
</tr>
<tr>
<td>apps</td>
<td></td>
</tr>
<tr>
<td>articles</td>
<td></td>
</tr>
<tr>
<td>files</td>
<td></td>
</tr>
<tr>
<td>groups</td>
<td></td>
</tr>
<tr>
<td>records</td>
<td></td>
</tr>
<tr>
<td>topics</td>
<td></td>
</tr>
<tr>
<td>users</td>
<td></td>
</tr>
<tr>
<td>ConnectApi. RecommendedObjectType</td>
<td>Type of object being recommended.</td>
</tr>
<tr>
<td>Today—Static recommendations that don’t have an ID, for example, the Today app recommendation.</td>
<td></td>
</tr>
<tr>
<td>ConnectApi. RecordColumnOrder</td>
<td>Order in which fields are rendered in a grid.</td>
</tr>
<tr>
<td>LeftRight—Fields are rendered from left to right.</td>
<td></td>
</tr>
<tr>
<td>TopDown—Fields are rendered from the top down.</td>
<td></td>
</tr>
<tr>
<td>ConnectApi. RecordFieldType</td>
<td>Data type of a record field.</td>
</tr>
<tr>
<td>Address</td>
<td></td>
</tr>
<tr>
<td>Blank</td>
<td></td>
</tr>
<tr>
<td>Boolean</td>
<td></td>
</tr>
<tr>
<td>Compound</td>
<td></td>
</tr>
<tr>
<td>CreatedBy</td>
<td></td>
</tr>
<tr>
<td>Date</td>
<td></td>
</tr>
<tr>
<td>DateTime</td>
<td></td>
</tr>
<tr>
<td>Email</td>
<td></td>
</tr>
<tr>
<td>LastModifiedBy</td>
<td></td>
</tr>
<tr>
<td>Location</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td></td>
</tr>
<tr>
<td>Number</td>
<td></td>
</tr>
<tr>
<td>Percent</td>
<td></td>
</tr>
<tr>
<td>Phone</td>
<td></td>
</tr>
<tr>
<td>Picklist</td>
<td></td>
</tr>
<tr>
<td>Enum</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td><strong>ConnectApi.RelatedFeedPostType</strong></td>
<td>Type of related feed post.</td>
</tr>
<tr>
<td>• <strong>Answered</strong>—Related questions that have at least one answer.</td>
<td></td>
</tr>
<tr>
<td>• <strong>BestAnswer</strong>—Related questions that have a best answer.</td>
<td></td>
</tr>
<tr>
<td>• <strong>Generic</strong>—All types of related questions, including answered, with a best answer, and unanswered.</td>
<td></td>
</tr>
<tr>
<td>• <strong>Unanswered</strong>—Related questions that don’t have answers.</td>
<td></td>
</tr>
<tr>
<td><strong>ConnectApi.SearchBoostBuryRuleAction</strong></td>
<td>Action of the boost and bury rule.</td>
</tr>
<tr>
<td>• <strong>Boost</strong>—Boost rule. Increases search result rankings for targeted products.</td>
<td></td>
</tr>
<tr>
<td>• <strong>Bury</strong>—Bury rule. Decreases search result rankings for targeted products.</td>
<td></td>
</tr>
<tr>
<td><strong>ConnectApi.SearchBoostBuryRuleOperation</strong></td>
<td>Operation for the conditions of the target expression in the boost and bury rule.</td>
</tr>
<tr>
<td>• <strong>AllOf</strong>—All-of operation.</td>
<td></td>
</tr>
<tr>
<td>• <strong>AnyOf</strong>—Any-of operation.</td>
<td></td>
</tr>
<tr>
<td><strong>ConnectApi.SegmentType</strong></td>
<td>Type of segment.</td>
</tr>
<tr>
<td>• <strong>Dbt</strong>—Data build tool</td>
<td></td>
</tr>
<tr>
<td><strong>ConnectApi.SellingModelType</strong></td>
<td>Type of product selling model.</td>
</tr>
<tr>
<td>• <strong>Evergreen</strong>—A subscription without an end date. An evergreen subscription continues until the shopper affirmatively cancels it.</td>
<td></td>
</tr>
<tr>
<td>• <strong>OneTime</strong>—A product that isn’t sold as a subscription.</td>
<td></td>
</tr>
<tr>
<td>• <strong>TermDefined</strong>—A subscription with a defined end date. The subscription continues for a specified time period. When the term ends, the subscription ends.</td>
<td></td>
</tr>
<tr>
<td><strong>ConnectApi.SitesPageType</strong></td>
<td>Type of site search result item.</td>
</tr>
<tr>
<td>• <strong>ContentPage</strong></td>
<td></td>
</tr>
<tr>
<td>• <strong>SitePage</strong></td>
<td></td>
</tr>
<tr>
<td><strong>ConnectApi.SocialNetworkProvider</strong></td>
<td>Social network provider.</td>
</tr>
<tr>
<td>• <strong>Facebook</strong></td>
<td></td>
</tr>
<tr>
<td>• <strong>GooglePlus</strong></td>
<td></td>
</tr>
<tr>
<td>• <strong>Instagram</strong></td>
<td></td>
</tr>
<tr>
<td>• <strong>InstagramBusiness</strong></td>
<td></td>
</tr>
<tr>
<td>• <strong>KakaoTalk</strong></td>
<td></td>
</tr>
<tr>
<td>• <strong>Kik</strong></td>
<td></td>
</tr>
<tr>
<td>• <strong>Line</strong></td>
<td></td>
</tr>
<tr>
<td>Enum</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>LinkedIn</td>
<td></td>
</tr>
<tr>
<td>Messenger</td>
<td></td>
</tr>
<tr>
<td>Other</td>
<td></td>
</tr>
<tr>
<td>Pinterest</td>
<td></td>
</tr>
<tr>
<td>QQ</td>
<td></td>
</tr>
<tr>
<td>Rypple</td>
<td></td>
</tr>
<tr>
<td>SinaWeibo</td>
<td></td>
</tr>
<tr>
<td>SMS</td>
<td></td>
</tr>
<tr>
<td>Snapchat</td>
<td></td>
</tr>
<tr>
<td>Telegram</td>
<td></td>
</tr>
<tr>
<td>Twitter</td>
<td></td>
</tr>
<tr>
<td>VKontakte</td>
<td></td>
</tr>
<tr>
<td>WeChat</td>
<td></td>
</tr>
<tr>
<td>WhatsApp</td>
<td></td>
</tr>
<tr>
<td>YouTube</td>
<td></td>
</tr>
</tbody>
</table>

**ConnectApi.SocialPostMessageType**  
Message type of the social post.

- Comment
- Direct
- Post
- PrivateMessage
- Reply
- Retweet
- Tweet

**ConnectApi.SocialPostStatusType**  
State of the social post.

- ApprovalPending
- ApprovalRecalled
- ApprovalRejected
- Deleted
- Failed
- Hidden
- Pending
- Sent
- Unknown

**ConnectApi.SortOrder**  
Order for sorting.

- Ascending—Items are in ascending alphabetical order (A-Z).
- Descending—Items are in descending alphabetical order (Z-A).
<table>
<thead>
<tr>
<th><strong>Enum</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>ConnectApi.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>ConnectApi.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>ConnectApi.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>ConnectApi.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>ConnectApi.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>ConnectApi.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2041
<table>
<thead>
<tr>
<th>Enum</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConnectApi.UserMission</td>
<td>Type of mission activity for a user.</td>
</tr>
<tr>
<td>ActivityType</td>
<td></td>
</tr>
<tr>
<td>FeedItemAnswerAQuestion</td>
<td>User answered a question.</td>
</tr>
<tr>
<td>FeedItemLikeSomething</td>
<td>User liked a post or comment.</td>
</tr>
<tr>
<td>FeedItemMarkAnswerAsBest</td>
<td>User marked an answer as the best answer.</td>
</tr>
<tr>
<td>FeedItemPostQuestion</td>
<td>User posted a question.</td>
</tr>
<tr>
<td>FeedItemReceiveAComment</td>
<td>User received a comment on a post.</td>
</tr>
<tr>
<td>FeedItemReceiveALike</td>
<td>User received a like on a post or comment.</td>
</tr>
<tr>
<td>FeedItemReceiveAnAnswer</td>
<td>User received an answer to a question.</td>
</tr>
<tr>
<td>FeedItemWriteAComment</td>
<td>User commented on a post.</td>
</tr>
<tr>
<td>FeedItemWriteAPost</td>
<td>User made a post.</td>
</tr>
<tr>
<td>FeedItemYourAnswerMarkedBest</td>
<td>User’s answer was marked as the best answer.</td>
</tr>
<tr>
<td>ConnectApi.UserProfile</td>
<td>Type of user profile tab.</td>
</tr>
<tr>
<td>TabType</td>
<td></td>
</tr>
<tr>
<td>CustomVisualForce</td>
<td>Tab that displays data from a Visualforce page.</td>
</tr>
<tr>
<td>CustomWeb</td>
<td>Tab that displays data from any external web-based application or web page.</td>
</tr>
<tr>
<td>Element</td>
<td>Tab that displays generic content inline.</td>
</tr>
<tr>
<td>Feed</td>
<td>Tab that displays the Chatter feed.</td>
</tr>
<tr>
<td>Overview</td>
<td>Tab that displays user details.</td>
</tr>
<tr>
<td>ConnectApi.UserType</td>
<td>Type of user.</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>ChatterGuest</td>
<td>User is an external user in a private group.</td>
</tr>
<tr>
<td>ChatterOnly</td>
<td>User is a Chatter Free customer.</td>
</tr>
<tr>
<td>Guest</td>
<td>User is unauthenticated.</td>
</tr>
<tr>
<td>Internal</td>
<td>User is a standard org member.</td>
</tr>
<tr>
<td>Portal</td>
<td>User is an external user in an Experience Cloud site.</td>
</tr>
<tr>
<td>System</td>
<td>User is Chatter Expert or a system user.</td>
</tr>
<tr>
<td>Undefined</td>
<td>User is a user type that is a custom object.</td>
</tr>
<tr>
<td>ConnectApi.WishlistItem</td>
<td>Sort order for wishlist items.</td>
</tr>
<tr>
<td>SortOrder</td>
<td></td>
</tr>
<tr>
<td>CreatedDateAsc</td>
<td>Sorts by oldest creation date.</td>
</tr>
<tr>
<td>CreatedDateDesc</td>
<td>Sorts by most recent creation date.</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Approved</td>
<td></td>
</tr>
<tr>
<td>Fault</td>
<td></td>
</tr>
<tr>
<td>Held</td>
<td></td>
</tr>
<tr>
<td>NoResponse</td>
<td></td>
</tr>
<tr>
<td>Pending</td>
<td></td>
</tr>
<tr>
<td>Reassigned</td>
<td></td>
</tr>
</tbody>
</table>
Rejected
• Removed
• Started

**ConnectApi.ZoneSearchResultType**
Zone search result type.
• Article—Search results contain only articles.
• Question—Search results contain only questions.

**ConnectApi.ZoneShowIn**
Zone search result location.
• Community—Available in an Experience Cloud site.
• Internal—Available internally only.
• Portal—Available in a portal.

---

**ConnectApi Exceptions**

The ConnectApi namespace contains exception classes.
All exceptions classes support built-in methods for returning the error message and exception type. See [Exception Class and Built-In Exceptions](#) on page 3076.

The ConnectApi namespace contains these exceptions:

<table>
<thead>
<tr>
<th>Exception</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConnectApi.ConnectApiException</td>
<td>Any logic error in the way your application is utilizing ConnectApi code. This is equivalent to receiving a 400 error from Connect REST API.</td>
</tr>
<tr>
<td>ConnectApi.NotFoundException</td>
<td>Any issues with the specified resource being found. This is equivalent to receiving a 404 error from Connect REST API.</td>
</tr>
<tr>
<td>ConnectApi.RateLimitException</td>
<td>When you exceed the rate limit. This is equivalent to receiving a 503 Service Unavailable error from Connect REST API.</td>
</tr>
</tbody>
</table>

**ConnectApi Utilities**

The ConnectApi namespace contains a utility class.

<table>
<thead>
<tr>
<th>Utility</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConnectApi.ConnectUtilities.unwrapApexWrapper()</td>
<td>Reserved for internal use.</td>
</tr>
</tbody>
</table>

**ConnectApi Release Notes**

Use the Salesforce Release Notes to learn about the most recent updates and changes to the ConnectApi namespace in Apex.
For updates and changes that impact Apex, including ConnectApi, see the Apex Release Notes.

For new and changed ConnectApi classes and enums, see ConnectApi (Connect in Apex): New and Changed Classes and Enums in the Salesforce Release Notes.

Database Namespace

The Database namespace provides classes used with DML operations.

The following are the classes in the Database namespace.

IN THIS SECTION:

Batchable Interface
The class that implements this interface can be executed as a batch Apex job.

BatchableContext Interface
Represents the parameter type of a batch job method and contains the batch job ID. This interface is implemented internally by Apex.

DeletedRecord Class
Contains information about a deleted record.

DeleteResult Class
Represents the result of a delete DML operation returned by the Database.delete method.

DMLOptions Class
Enables you to set options related to DML operations.

DMLOptions.AssignmentRuleHeader Class
Enables setting assignment rule options.

DMLOptions.DuplicateRuleHeader Class
Determines options for using duplicate rules to detect duplicate records. Duplicate rules are part of the Duplicate Management feature.

DMLOptions.EmailHeader Class
Enables setting email options.

DuplicateError Class
Contains information about an error that occurred when an attempt was made to save a duplicate record. Use if your organization has set up duplicate rules, which are part of the Duplicate Management feature.

EmptyRecycleBinResult Class
The result of the emptyRecycleBin DML operation returned by the Database.emptyRecycleBin method.

Error Class
Represents information about an error that occurred during a DML operation when using a Database method.

GetDeletedResult Class
Contains the deleted records retrieved for a specific sObject type and time window.

GetUpdatedResult Class
Contains the result for the Database.getUpdated method call.

LeadConvert Class
Contains information used for lead conversion.
LeadConvertResult Class
The result of a lead conversion.

MergeResult Class
Contains the result of a merge Database method operation.

QueryLocator Class
Represents the record set returned by Database.getQueryLocator and used with Batch Apex.

QueryLocatorIterator Class
Represents an iterator over a query locator record set.

SaveResult Class
The result of an insert or update DML operation returned by a Database method.

UndeleteResult Class
The result of an undelete DML operation returned by the Database.undelete method.

UpsertResult Class
The result of an upsert DML operation returned by the Database.upsert method.

Batchable Interface
The class that implements this interface can be executed as a batch Apex job.

Namespace
Database

SEE ALSO:
Apex Developer Guide: Using Batch

Batchable Methods
The following are methods for Batchable.

IN THIS SECTION:
execute(jobId, recordList)
Gets invoked when the batch job executes and operates on one batch of records. Contains or calls the main execution logic for the batch job.

finish(jobId)
Gets invoked when the batch job finishes. Place any cleanup code in this method.

start(jobId)
Gets invoked when the batch job starts. Returns the record set as an iterable that will be batched for execution.

start(jobId)
Gets invoked when the batch job starts. Returns the record set as a QueryLocator object that will be batched for execution.
execute(jobId, recordList)
Gets invoked when the batch job executes and operates on one batch of records. Contains or calls the main execution logic for the batch job.

Signature
public Void execute(Database.BatchableContext jobId, List<sObject> recordList)

Parameters
jobId
  Type: Database.BatchableContext
  Contains the job ID.
recordList
  Type: List<sObject>
  Contains the batch of records to process.

Return Value
Type: Void

finish(jobId)
Gets invoked when the batch job finishes. Place any clean up code in this method.

Signature
public Void finish(Database.BatchableContext jobId)

Parameters
jobId
  Type: Database.BatchableContext
  Contains the job ID.

Return Value
Type: Void

start(jobId)
Gets invoked when the batch job starts. Returns the record set as an iterable that will be batched for execution.

Signature
public System.Iterable start(Database.BatchableContext jobId)
Parameters

jobId
Type: Database.BatchableContext
Contains the job ID.

Return Value
Type: System.Iterable

start(jobId)
Gets invoked when the batch job starts. Returns the record set as a QueryLocator object that will be batched for execution.

Signature

public Database.QueryLocator start(Database.BatchableContext jobId)

Parameters

jobId
Type: Database.BatchableContext
Contains the job ID.

Return Value
Type: Database.QueryLocator

BatchableContext Interface

Represents the parameter type of a batch job method and contains the batch job ID. This interface is implemented internally by Apex.

Namespace

Database

SEE ALSO:
Batchable Interface

BatchableContext Methods

The following are methods for BatchableContext.

IN THIS SECTION:

getChildJobId()
Returns the ID of the current batch job chunk that is being processed.
getJobId()
Returns the batch job ID.

getChildJobId()
Returns the ID of the current batch job chunk that is being processed.

Signature
public Id getChildJobId()

Return Value
Type: ID

getJobId()
Returns the batch job ID.

Signature
public Id getJobId()

Return Value
Type: ID

DeletedRecord Class
Contains information about a deleted record.

Namespace
Database

Usage
The getDeletedRecords method of the Database.GetDeletedResult class returns a list of Database.DeletedRecord objects. Use the methods in the Database.DeletedRecord class to retrieve details about each deleted record.

DeletedRecord Methods
The following are methods for DeletedRecord. All are instance methods.

IN THIS SECTION:
getDeletedDate()
Returns the deleted date of this record.
getId()  
Returns the ID of a record deleted within the time window specified in the Database.getDeleted method.

getDeletedDate()  
Returns the deleted date of this record.

**Signature**  
`public Date getDeletedDate()`  

**Return Value**  
Type: Date  

getId()  
Returns the ID of a record deleted within the time window specified in the Database.getDeleted method.

**Signature**  
`public Id getId()`  

**Return Value**  
Type: ID  

### DeleteResult Class

Represents the result of a delete DML operation returned by the Database.delete method.

**Namespace**  
Database

**Usage**  
An array of Database.DeleteResult objects is returned with the delete database method. Each element in the DeleteResult array corresponds to the sObject array passed as the sObject[] parameter in the delete Database method; that is, the first element in the DeleteResult array matches the first element passed in the sObject array, the second element corresponds with the second element, and so on. If only one sObject is passed in, the DeleteResult array contains a single element.

**Example**  
The following example shows how to obtain and iterate through the returned Database.DeleteResult objects. It deletes some queried accounts using Database.delete with a false second parameter to allow partial processing of records on failure. Next, it
iterates through the results to determine whether the operation was successful or not for each record. It writes the ID of every record that was processed successfully to the debug log, or error messages and fields of the failed records.

```java
// Query the accounts to delete
Account[] accts = [SELECT Id from Account WHERE Name LIKE 'Acme%'];

// Delete the accounts
Database.DeleteResult[] drList = Database.delete(accts, false);

// Iterate through each returned result
for(Database.DeleteResult dr : drList) {
    if (dr.isSuccess()) {
        // Operation was successful, so get the ID of the record that was processed
        System.debug('Successfully deleted account with ID: ' + dr.getId());
    } else {
        // Operation failed, so get all errors
        for(Database.Error err : dr.getErrors()) {
            System.debug('The following error has occurred: ');
            System.debug(err.getStatusCode() + ': ' + err.getMessage());
            System.debug('Account fields that affected this error: ' + err.getFields());
        }
    }
}
```

### DeleteResult Methods

The following are methods for `DeleteResult`. All are instance methods.

**getErrors()**

If an error occurred, returns an array of one or more database error objects providing the error code and description. If no error occurred, returns an empty set.

**getErrId()**

Returns the ID of the sObject you were trying to delete.

**isSuccess()**

A Boolean value that is set to `true` if the DML operation was successful for this object, `false` otherwise.

**getErrors()**

If an error occurred, returns an array of one or more database error objects providing the error code and description. If no error occurred, returns an empty set.

**Signature**

```
public Database.Error[] getErrors()
```

**Return Value**

Type: `Database.Error[]`
getId()
Returns the ID of the sObject you were trying to delete.

Signature
public ID getId()

Return Value
Type: ID

isSuccess()
A Boolean value that is set to true if the DML operation was successful for this object, false otherwise.

Signature
public Boolean isSuccess()

Return Value
Type: Boolean

DMLOptions Class
Enables you to set options related to DML operations.

Namespace
Database

Usage
Database.DMLOptions is only available for Apex saved against API versions 15.0 and higher. DMLOptions settings take effect only for record operations performed using Apex DML and not through the Salesforce user interface. The DMLOptions class has three child options.

DML Child Options
- DmlOptions.AssignmentRuleHeader—Enables setting assignment rule options.
- DmlOptions.DuplicateRuleHeader—Determines options for using duplicate rules to detect duplicate records. Duplicate rules are part of the Duplicate Management feature.
- DmlOptions.EmailHeader—Enables setting email options.

DmlOptions Properties
The following are properties for DmlOptions.
allowFieldTruncation
Specifies the truncation behavior of large strings.

Signature
public Boolean allowFieldTruncation {get; set;}

Property Value
Type: Boolean

Usage
In Apex saved against API versions previous to 15.0, if you specify a value for a string and that value is too large, the value is truncated. For API version 15.0 and later, if a value is specified that is too large, the operation fails and an error message is returned. The allowFieldTruncation property allows you to specify that the previous behavior, truncation, be used instead of the new behavior in Apex saved against API versions 15.0 and later.

assignmentRuleHeader
Specifies the assignment rule to be used when creating a case or lead.

Signature
public Database.DmlOptions.AssignmentRuleHeader assignmentRuleHeader {get; set;}

Property Value
Type: Database.DMLOptions.AssignmentRuleHeader

Usage
Note: The Database.DMLOptions object supports assignment rules for cases and leads, but not for accounts.
**emailHeader**
Specifies additional information regarding the automatic email that gets sent when an event occurs.

**Signature**
```csharp
public Database.DmlOptions.EmailHeader emailHeader {get; set;}
```

**Property Value**
Type: `Database.DmlOptions.EmailHeader`

**Usage**
The Salesforce user interface allows you to specify whether or not to send an email when the following events occur.

- Creation of a new case or task
- Conversion of a case email to a contact
- New user email notification
- Lead queue email notification
- Password reset

In Apex saved against API version 15.0 or later, the `Database.DmlOptions.emailHeader` property enables you to specify additional information regarding the email that gets sent when one of the events occurs because of the code's execution.

**localeOptions**
Specifies the language of any labels that are returned by Apex.

**Signature**
```csharp
public Database.DmlOptions.LocaleOptions localeOptions {get; set;}
```

**Property Value**
Type: `Database.DmlOptions.LocaleOptions`

**Usage**
The value must be a valid user locale (language and country), such as de_DE or en_GB. The value is a String, 2-5 characters long. The first two characters are always an ISO language code, for example `fr` or `en`. If the value is further qualified by a country, then the string also has an underscore (_) and another ISO country code, for example `US` or `UK`. For example, the string for the United States is `en_US`, and the string for French Canadian is `fr_CA`.

**optAllOrNone**
Specifies whether the operation allows for partial success.

**Signature**
```csharp
public Boolean optAllOrNone {get; set;}
```
Property Value

Type: Boolean

Usage

If `optAllOrNone` is set to `true`, all changes are rolled back if any record causes errors. The default for this property is `false` and successfully processed records are committed while records with errors aren't.

This property is available in Apex saved against Salesforce API version 20.0 and later.

DmlOptions.AssignmentRuleHeader Class

Enables setting assignment rule options.

Namespace

Database

Example

The following example uses the `useDefaultRule` option:

```java
Database.DMLOptions dmo = new Database.DMLOptions();
dmo.assignmentRuleHeader.useDefaultRule = true;

Lead l = new Lead(company='ABC', lastname='Smith');
l.setOptions(dmo);
insert l;
```

The following example uses the `assignmentRuleId` option:

```java
Database.DMLOptions dmo = new Database.DMLOptions();
dmo.assignmentRuleHeader.assignmentRuleId = '01QD0000000EqAn';

Lead l = new Lead(company='ABC', lastname='Smith');
l.setOptions(dmo);
insert l;
```

DmlOptions.AssignmentRuleHeader Properties

The following are properties for `DmlOptions.AssignmentRuleHeader`.

IN THIS SECTION:

- `assignmentRuleId`
  - Specifies the ID of a specific assignment rule to run for the case or lead. The assignment rule can be active or inactive.

- `useDefaultRule`
  - If specified as `true` for a case or lead, the system uses the default (active) assignment rule for the case or lead. If specified, do not specify an `assignmentRuleId`. 
**assignmentRuleID**
Specifies the ID of a specific assignment rule to run for the case or lead. The assignment rule can be active or inactive.

**Signature**
```csharp
public Id assignmentRuleID {get; set;}
```

**Property Value**
Type: ID

**Usage**
The ID can be retrieved by querying the AssignmentRule sObject. If specified, do not specify `useDefaultRule`.
If the value is not in the correct ID format (15-character or 18-character Salesforce ID), the call fails and an exception is returned.

**useDefaultRule**
If specified as `true` for a case or lead, the system uses the default (active) assignment rule for the case or lead. If specified, do not specify an `assignmentRuleId`.

**Signature**
```csharp
public Boolean useDefaultRule {get; set;}
```

**Property Value**
Type: Boolean

**Usage**
If there are no assignment rules in the organization, in API version 29.0 and earlier, creating a case or lead with `useDefaultRule` set to `true` results in the case or lead being assigned to the predefined default owner. In API version 30.0 and later, the case or lead is unassigned and doesn’t get assigned to the default owner.

**DMLOptions.DuplicateRuleHeader Class**
Determines options for using duplicate rules to detect duplicate records. Duplicate rules are part of the Duplicate Management feature.

**Namespace**
```csharp
Database
```
Example
The following example shows how to save an account record that’s been identified as a duplicate. To learn how to iterate through duplicate errors, see DuplicateError Class.

```apex
Database.DMLOptions dml = new Database.DMLOptions();
dml.DuplicateRuleHeader.allowSave = true;
dml.DuplicateRuleHeader.runAsCurrentUser = true;
Account duplicateAccount = new Account(Name='dupe');
Database.SaveResult sr = Database.insert(duplicateAccount, dml);
if (sr.isSuccess()) {
    System.debug('Duplicate account has been inserted in Salesforce!');
}
```

IN THIS SECTION:
DMLOptions.DuplicateRuleHeader Properties

DMLOptions.DuplicateRuleHeader Properties
The following are properties for DMLOptions.DuplicateRuleHeader.

IN THIS SECTION:
allowSave
For a duplicate rule, when the Alert option is enabled, bypass alerts and save duplicate records by setting this property to true. Prevent duplicate records from being saved by setting this property to false.

runAsCurrentUser
Make sure that sharing rules for the current user are enforced when duplicate rules run by setting this property to true. Use the sharing rules specified in the class for the request by setting this property to false. If no sharing rules are specified, Apex code runs in system context and sharing rules for the current user are not enforced.

**allowSave**
For a duplicate rule, when the Alert option is enabled, bypass alerts and save duplicate records by setting this property to true. Prevent duplicate records from being saved by setting this property to false.

**Signature**
public Boolean allowSave {get; set;}

**Property Value**
Type: Boolean
Example

This example shows how to save an account record that’s been identified as a duplicate. 

```java
Database.DMLOptions dml = new Database.DMLOptions();
dml.DuplicateRuleHeader.allowSave = true;
dml.DuplicateRuleHeader.runAsCurrentUser = true;
Account duplicateAccount = new Account(Name='dupe');
Database.SaveChanges sr = Database.insert(duplicateAccount, dml);
if (sr.isSuccess()) {
    System.debug('Duplicate account has been inserted in Salesforce!');
}
```

runAsCurrentUser

Make sure that sharing rules for the current user are enforced when duplicate rules run by setting this property to `true`. Use the sharing rules specified in the class for the request by setting this property to `false`. If no sharing rules are specified, Apex code runs in system context and sharing rules for the current user are not enforced.

Signature

`public Boolean runAsCurrentUser {get; set;}`

Property Value

Type: Boolean

Usage

If specified as `true`, duplicate rules run for the current user, which ensures users can’t view duplicate records that aren’t available to them.

Use `runAsCurrentUser = true` to detect duplicates when converting leads to contacts. Typically, lead conversion Apex code runs in a system context and does not enforce sharing rules for the current user.

Example

This example shows how to set options so that duplicate rules run for the current user when saving a new account.

```java
Database.DMLOptions dml = new Database.DMLOptions();
dml.DuplicateRuleHeader.allowSave = true;
dml.DuplicateRuleHeader.runAsCurrentUser = true;
Account duplicateAccount = new Account(Name='dupe');
Database.SaveChanges sr = Database.insert(duplicateAccount, dml);
if (sr.isSuccess()) {
    System.debug('Duplicate account has been inserted in Salesforce!');
}
```
Namespace

Database

Usage

Even though auto-sent emails can be triggered by actions in the Salesforce user interface, the DMLOptions settings for emailHeader take effect only for DML operations carried out in Apex code.

Example

In the following example, the triggerAutoResponseEmail option is specified:

```java
Account a = new Account(name='Acme Plumbing');

    insert a;

    Contact c = new Contact(email='jplumber@salesforce.com',
        firstname='Joe', lastname='Plumber', accountid=a.id);

    insert c;

Database.DMLOptions dlo = new Database.DMLOptions();

dlo.EmailHeader.triggerAutoResponseEmail = true;

Case ca = new Case(subject='Plumbing Problems', contactid=c.id);

database.insert(ca, dlo);
```

Suppose that you use an after-insert or after-update trigger to change ownership of leads, contacts, or opportunities. If you use the API to change record ownership, or if a Lightning Experience user changes a record’s owner, no email notification is sent. To send email notifications to a record’s new owner, set the triggerUserEmail property to true.

DmLOptions.EmailHeader Properties

The following are properties for DmLOptions.EmailHeader.

IN THIS SECTION:

- triggerAutoResponseEmail
  Indicates whether to trigger auto-response rules (true) or not (false), for leads and cases.

- triggerOtherEmail
  Indicates whether to trigger email outside the organization (true) or not (false).

- triggerUserEmail
  Indicates whether to trigger email that is sent to users in the organization (true) or not (false).

**triggerAutoResponseEmail**

Indicates whether to trigger auto-response rules (true) or not (false), for leads and cases.
Signature
public Boolean triggerAutoResponseEmail {get; set;}

Property Value
Type: Boolean

Usage
This email can be automatically triggered by a number of events, for example creating a case or resetting a user password. If this value is set to true, when a case is created, if there is an email address for the contact specified in ContactID, the email is sent to that address. If not, the email is sent to the address specified in SuppliedEmail

triggerOtherEmail
Indicates whether to trigger email outside the organization (true) or not (false).

Signature
public Boolean triggerOtherEmail {get; set;}

Property Value
Type: Boolean

Usage
This email can be automatically triggered by creating, editing, or deleting a contact for a case.

Note: Email sent through Apex because of a group event includes additional behaviors. A group event is an event for which IsGroupEvent is true. The EventAttendee object tracks the users, leads, or contacts that are invited to a group event. Note the following behaviors for group event email sent through Apex:
- Sending a group event invitation to a lead or contact respects the triggerOtherEmail option
- Email sent when updating or deleting a group event also respects the triggerUserEmail and triggerOtherEmail options, as appropriate

triggerUserEmail
Indicates whether to trigger email that is sent to users in the organization (true) or not (false).

Signature
public Boolean triggerUserEmail {get; set;}

Property Value
Type: Boolean
Usage

This email can be automatically triggered by a number of events; resetting a password, creating a new user, or creating or modifying a task.

**Note:** Adding comments to a case in Apex doesn’t trigger email to users in the organization even if triggerUserEmail is set to `true`.

**Note:** Email sent through Apex because of a group event includes additional behaviors. A group event is an event for which `IsGroupEvent` is true. The EventAttendee object tracks the users, leads, or contacts that are invited to a group event. Note the following behaviors for group event email sent through Apex:

- Sending a group event invitation to a user respects the triggerUserEmail option
- Email sent when updating or deleting a group event also respects the triggerUserEmail and triggerOtherEmail options, as appropriate

DuplicateError Class

Contains information about an error that occurred when an attempt was made to save a duplicate record. Use if your organization has set up duplicate rules, which are part of the Duplicate Management feature.

Namespace

Database

Example

When you try to save a record that’s identified as a duplicate record by a duplicate rule, you’ll receive a duplicate error. If the duplicate rule contains the Allow action, an attempt will be made to bypass the error.

```java
// Try to save a duplicate account
Account duplicateAccount = new Account(Name='Acme', BillingCity='San Francisco');
Database.SaveResult sr = Database.insert(duplicateAccount, false);
if (!sr.isSuccess()) {
    // Insertion failed due to duplicate detected
    for(Database.Error duplicateError : sr.getErrors()) {
        Datacloud.DuplicateResult duplicateResult = ((Database.DuplicateError)duplicateError).getDuplicateResult();
        System.debug('Duplicate records have been detected by ' + duplicateResult.getDuplicateRule());
        System.debug(duplicateResult.getErrorMessage());
    }
    // If the duplicate rule is an alert rule, we can try to bypass it
    Database.DMLOptions dml = new Database.DMLOptions();
    dml.DuplicateRuleHeader.AllowSave = true;
    Database.SaveResult sr2 = Database.insert(duplicateAccount, dml);
    if (sr2.isSuccess()) {
        System.debug('Duplicate account has been inserted in Salesforce!');
    }
}
```
DuplicateError Methods

The following are methods for DuplicateError.

getDuplicateResult()
Returns the details of a duplicate rule and duplicate records found by the duplicate rule.

getFields()
Returns an array of one or more field names. Identifies which fields in the object, if any, affected the error condition.

getMessage()
Returns the error message text.

getStatusCode()
Returns a code that characterizes the error.

getDuplicateResult()

Returns the details of a duplicate rule and duplicate records found by the duplicate rule.

Signature

public Datacloud.DuplicateResult getDuplicateResult()

Return Value

Type: Datacloud.DuplicateResult

Example

This example shows the code used to get the possible duplicates and related match information after saving a new contact. This code is part of a custom application that implements duplicate management when users add a contact. See DuplicateResult Class on page 2092 to check out the entire sample application.

Datacloud.DuplicateResult duplicateResult =
   duplicateError.getDuplicateResult();

getFields()

Returns an array of one or more field names. Identifies which fields in the object, if any, affected the error condition.
**Signature**

```java
public List<String> getFields()
```

**Return Value**

Type: `List<String>`

---

**getMesage()**

Returns the error message text.

**Signature**

```java
public String getMessage()
```

**Return Value**

Type: `String`

---

**getStatusCode()**

Returns a code that characterizes the error.

**Signature**

```java
public StatusCode getStatusCode()
```

**Return Value**

Type: `StatusCode`

---

**EmptyRecycleBinResult Class**

The result of the emptyRecycleBin DML operation returned by the `Database.emptyRecycleBin` method.

**Namespace**

`Database`

**Usage**

A list of `Database.EmptyRecycleBinResult` objects is returned by the `Database.emptyRecycleBin` method. Each object in the list corresponds to either a record ID or an sObject passed as the parameter in the `Database.emptyRecycleBin` method. The first index in the `EmptyRecycleBinResult` list matches the first record or sObject specified in the list, the second with the second, and so on.

**EmptyRecycleBinResult Methods**

The following are methods for `EmptyRecycleBinResult`. All are instance methods.
IN THIS SECTION:

getErrors()
If an error occurred during the delete for this record or sObject, returns a list of one or more Database.Error objects. If no errors occurred, the returned list is empty.

getId()
Returns the ID of the record or sObject you attempted to delete.

isSuccess()
Returns true if the record or sObject was successfully removed from the Recycle Bin; otherwise false.

getErrors()
If an error occurred during the delete for this record or sObject, returns a list of one or more Database.Error objects. If no errors occurred, the returned list is empty.

Signature
public Database.Errors[] getErrors()

Return Value
Type: Database.Errors []

getId()
Returns the ID of the record or sObject you attempted to delete.

Signature
public ID getId()

Return Value
Type: ID

isSuccess()
Returns true if the record or sObject was successfully removed from the Recycle Bin; otherwise false.

Signature
public Boolean isSuccess()

Return Value
Type: Boolean

Error Class
Represents information about an error that occurred during a DML operation when using a Database method.
Namespace

Database

Usage

Error class is part of SaveResult, which is generated when a user attempts to save a Salesforce record.

SEE ALSO:

SaveResult Class
DuplicateError Class

Error Methods

The following are methods for Error. All are instance methods.

IN THIS SECTION:

getFields()
Returns an array of one or more field names. Identifies which fields in the object, if any, affected the error condition.

getMessage()
Returns the error message text.

getStatusCode()
Returns a code that characterizes the error.

getFields()

Returns an array of one or more field names. Identifies which fields in the object, if any, affected the error condition.

Signature

public String[] getFields()

Return Value

Type: String[]

getMessage()

Returns the error message text.

Signature

public String getMessage()

Return Value

Type: String
getStatusCode()
Returns a code that characterizes the error.

Signature
public StatusCode getStatusCode()

Return Value
Type: StatusCode

Usage
The full list of status codes is available in the WSDL file for your organization (see Downloading Salesforce WSDLs and Client Authentication Certificates in the Salesforce online help.)

GetDeletedResult Class
Contains the deleted records retrieved for a specific sObject type and time window.

Namespace
Database

Usage
The Database.getDeleted method returns the deleted record information as a Database.GetDeletedResult object.

GetDeletedResult Methods
The following are methods for GetDeletedResult. All are instance methods.

IN THIS SECTION:
  getDeletedRecords()
  Returns a list of deleted records for the time window specified in the Database.getDeleted method call.
  getEarliestDateAvailable()
  Returns the date in Coordinated Universal Time (UTC) of the earliest physically deleted object for the sObject type specified in Database.getDeleted.
  getLatestDateCovered()
  Returns the date in Coordinated Universal Time (UTC) of the last date covered in the Database.getDeleted call.

getDeletedRecords()
Returns a list of deleted records for the time window specified in the Database.getDeleted method call.
Signature
public List<Database.DeletedRecord> getDeletedRecords()

Return Value
Type: List<Database.DeletedRecord>

getEarliestDateAvailable()
Returns the date in Coordinated Universal Time (UTC) of the earliest physically deleted object for the sObject type specified in Database.getDeleted.

Signature
public Date getEarliestDateAvailable()

Return Value
Type: Date

getLatestDateCovered()
Returns the date in Coordinated Universal Time (UTC) of the last date covered in the Database.getDeleted call.

Signature
public Date getLatestDateCovered()

Return Value
Type: Date

Usage
If there is a value, it is less than or equal to the endDate argument of Database.getDeleted. A value here indicates that, for safety, you should use this value for the startDate of your next call to capture the changes that started after this date but didn’t complete before endDate and were, therefore, not returned in the previous call.

GetUpdatedResult Class
Contains the result for the Database.getUpdated method call.

Namespace
Database
Usage

Use the methods in this class to obtain detailed information about the updated records returned by Database.getUpdated for a specific time window.

GetUpdatedResult Methods

The following are methods for GetUpdatedResult. All are instance methods.

IN THIS SECTION:
- getIds()<br>
  Returns the IDs of records updated within the time window specified in the Database.getUpdated method.<br>
- getLatestDateCovered()<br>
  Returns the date in Coordinated Universal Time (UTC) of the last date covered in the Database.getUpdated call.

getIds()

Returns the IDs of records updated within the time window specified in the Database.getUpdated method.

Signature

public List<Id> getIds()

Return Value

Type: List<ID>

getLatestDateCovered()

Returns the date in Coordinated Universal Time (UTC) of the last date covered in the Database.getUpdated call.

Signature

public Date getLatestDateCovered()

Return Value

Type: Date

LeadConvert Class

Contains information used for lead conversion.

Namespace

Database
Usage

The `convertLead` Database method converts a lead into an account and contact or an account and person account, as well as (optionally) an opportunity. The `convertLead` takes an instance of the `Database.LeadConvert` class as a parameter. Create an instance of this class and set the information required for conversion, such as setting the lead, and destination account and contact.

Note: The `Database.convertLead()` method can take one `LeadConvert` object or a list of `LeadConvert` objects.

Example

This example shows how to use the `Database.convertLead` method to convert a lead. It inserts a new lead, creates a `LeadConvert` object, sets its status to converted, then passes it to the `Database.convertLead` method. Finally, it verifies that the conversion was successful.

```java
Lead myLead = new Lead(LastName = 'Fry', Company='Fry And Sons');
insert myLead;

Database.LeadConvert lc = new Database.LeadConvert();
lc.setLeadId(myLead.id);

LeadStatus convertStatus = [SELECT Id, MasterLabel FROM LeadStatus WHERE IsConverted=true LIMIT 1];
lc.setConvertedStatus(convertStatus.MasterLabel);

Database.LeadConvertResult lcr = Database.convertLead(lc);
System.assert(lcr.isSuccess());
```

LeadConvert Constructors

The following are constructors for `LeadConvert`.

LeadConvert()

Creates a new instance of the `Database.LeadConvert` class.

Signature

`public LeadConvert()`
LeadConvert Methods

The following are methods for LeadConvert. All are instance methods.

IN THIS SECTION:

- `getAccountId()`
  Gets the ID of the account into which the lead will be merged.

- `getContactId()`
  Gets the ID of the contact into which the lead will be merged.

- `getConvertedStatus()`
  Gets the lead status value for a converted lead.

- `getLeadID()`
  Gets the ID of the lead to convert.

- `getOpportunityId()`
  Gets the ID of the existing opportunity that will be related to the resulting contact.

- `getOpportunityName()`
  Gets the name of the opportunity to create.

- `getOwnerID()`
  Gets the ID of the person to own any newly created account, contact, and opportunity.

- `getRelatedPersonAccountId()`
  Gets the ID of the existing person account into which the lead will be converted.

- `getRelatedPersonAccountRecord()`
  Gets the entity record of the new person account into which the lead will be converted.

- `isDoNotCreateOpportunity()`
  Indicates whether an Opportunity is created during lead conversion (false, the default) or not (true).

- `isOverWriteLeadSource()`
  Indicates whether the LeadSource field on the target Contact object is overwritten with the contents of the LeadSource field in the source Lead object (true), or not (false, the default).

- `isSendNotificationEmail()`
  Indicates whether a notification email is sent to the owner specified by setOwnerId (true) or not (false, the default).

- `setAccountId(accountId)`
  Sets the ID of the account into which the lead is merged. This value is required only when updating an existing account, including person accounts.

- `setContactId(contactId)`
  Sets the ID of the contact into which the lead will be merged (this contact must be associated with the account specified with setAccountId, and setAccountId must be specified). This value is required only when updating an existing contact.

- `setConvertedStatus(status)`
  Sets the lead status value for a converted lead. This field is required.

- `setDoNotCreateOpportunity(createOpportunity)`
  Specifies whether to create an opportunity during lead conversion. The default value is false: opportunities are created by default. Set this flag to true only if you do not want to create an opportunity from the lead.
setLeadId(leadId)
Sets the ID of the lead to convert. This field is required.

setOpportunityId(opportunityId)
Sets the ID of the opportunity into which the lead is merged. This value is required only when updating an existing opportunity.

setOpportunityName(opportunityName)
Sets the name of the opportunity to create. If no name is specified, this value defaults to the company name of the lead.

setOverwriteLeadSource(overwriteLeadSource)
Specifies whether to overwrite the LeadSource field on the target contact object with the contents of the LeadSource field in the source lead object. The default value is false, to not overwrite the field. If you specify this as true, you must also specify setContactId for the target contact.

setOwnerId(ownerId)
Specifies the ID of the person to own any newly created account, contact, and opportunity. If the application does not specify this value, the owner of the new object will be the owner of the lead.

setRelatedPersonAccountId(relatedPersonAccountId)
Sets the ID of the existing person account into which to convert the lead. This value is required only when updating an existing person account.

setRelatedPersonAccountRecord(Entity relatedPersonAccountRecord)
Sets the entity record of the new person account into which to convert the lead.

setSendNotificationEmail(sendEmail)
Specifies whether to send a notification email to the owner specified by setOwnerId. The default value is false, that is, to not send email.

getAccountId()
Gets the ID of the account into which the lead will be merged.

Signature
public ID getAccountId()

Return Value
Type: ID

getContactId()
Gets the ID of the contact into which the lead will be merged.

Signature
public ID getContactId()

Return Value
Type: ID
**getConvertedStatus()**
Gets the lead status value for a converted lead.

**Signature**
public String getConvertedStatus()

**Return Value**
Type: String

**getLeadID()**
Gets the ID of the lead to convert.

**Signature**
public ID getLeadID()

**Return Value**
Type: ID

**getOpportunityId()**
Gets the ID of the existing opportunity that will be related to the resulting contact.

**Signature**
public ID getOpportunityId()

**Return Value**
Type: ID

**getOpportunityName()**
Gets the name of the opportunity to create.

**Signature**
public String getOpportunityName()

**Return Value**
Type: String

**getOwnerID()**
Gets the ID of the person to own any newly created account, contact, and opportunity.
**Signature**

```java
public ID getOwnerID()
```

**Return Value**

Type: ID

---

**getRelatedPersonAccountID()**

Gets the ID of the existing person account into which the lead will be converted.

**Signature**

```java
public ID getRelatedPersonAccountID()
```

**Return Value**

Type: ID

---

**getRelatedPersonAccountRecord()**

Gets the entity record of the new person account into which the lead will be converted.

**Signature**

```java
public ID getRelatedPersonAccountRecord()
```

**Return Value**

Type: ID

---

**isDoNotCreateOpportunity()**

Indicates whether an Opportunity is created during lead conversion (`false`, the default) or not (`true`).

**Signature**

```java
public Boolean isDoNotCreateOpportunity()
```

**Return Value**

Type: `Boolean`

---

**isOverWriteLeadSource()**

Indicates whether the `LeadSource` field on the target Contact object is overwritten with the contents of the `LeadSource` field in the source Lead object (`true`), or not (`false`, the default).
**Signature**

`public Boolean isOverWriteLeadSource()`

**Return Value**

Type: Boolean

**isSendNotificationEmail()**

Indicates whether a notification email is sent to the owner specified by `setOwnerId` (true) or not (false, the default).

**Signature**

`public Boolean isSendNotificationEmail()`

**Return Value**

Type: Boolean

**setAccountId(accountId)**

Sets the ID of the account into which the lead is merged. This value is required only when updating an existing account, including person accounts.

**Signature**

`public Void setAccountId(ID accountId)`

**Parameters**

`accountId`

Type: ID

**Return Value**

Type: Void

**setContactId(contactId)**

Sets the ID of the contact into which the lead will be merged (this contact must be associated with the account specified with `setAccountId`, and `setAccountId` must be specified). This value is required only when updating an existing contact.

**Signature**

`public Void setContactId(ID contactId)`

**Parameters**

`contactId`

Type: ID
Return Value
Type: Void

Usage
If setContactId is specified, then the application creates a new contact that is implicitly associated with the account. The contact name and other existing data are not overwritten (unless setOverwriteLeadSource is set to true, in which case only the LeadSource field is overwritten).

**Important:** If you are converting a lead into a person account, do not specify setContactId or an error will result. Specify only setAccountId of the person account.

**setConvertedStatus(status)**
Sets the lead status value for a converted lead. This field is required.

Signature
```java
public Void setConvertedStatus(String status)
```

Parameters
- **status**
  Type: String

Return Value
Type: Void

**setDoNotCreateOpportunity(createOpportunity)**
Specifies whether to create an opportunity during lead conversion. The default value is false: opportunities are created by default. Set this flag to true only if you do not want to create an opportunity from the lead.

Signature
```java
public Void setDoNotCreateOpportunity(Boolean createOpportunity)
```

Parameters
- **createOpportunity**
  Type: Boolean

Return Value
Type: Void

**setLeadId(leadId)**
Sets the ID of the lead to convert. This field is required.
Signature
public Void setLeadId(ID leadId)

Parameters
leadId
Type: ID

Return Value
Type: Void

setOpportunityId(opportunityId)
Sets the ID of the opportunity into which the lead is merged. This value is required only when updating an existing opportunity.

Signature
public Void setOpportunityId(ID opportunityId)

Parameters
opportunityId
Type: ID

Return Value
Type: Void

setOpportunityName(opportunityName)
Sets the name of the opportunity to create. If no name is specified, this value defaults to the company name of the lead.

Signature
public Void setOpportunityName(String opportunityName)

Parameters
opportunityName
Type: String

Return Value
Type: Void

Usage
The maximum length of this field is 80 characters.
If `setDoNotCreateOpportunity` is true, no Opportunity is created and this field must be left blank; otherwise, an error is returned.

**setOverwriteLeadSource (overwriteLeadSource)**

Specifies whether to overwrite the `LeadSource` field on the target contact object with the contents of the `LeadSource` field in the source lead object. The default value is `false`, to not overwrite the field. If you specify this as `true`, you must also specify `setContactId` for the target contact.

**Signature**

```java
public Void setOverwriteLeadSource(Boolean overwriteLeadSource)
```

**Parameters**

- `overwriteLeadSource`  
  Type: `Boolean`

**Return Value**

Type: `Void`

**setOwnerId (ownerId)**

Specifies the ID of the person to own any newly created account, contact, and opportunity. If the application does not specify this value, the owner of the new object will be the owner of the lead.

**Signature**

```java
public Void setOwnerId(ID ownerId)
```

**Parameters**

- `ownerId`  
  Type: `ID`

**Return Value**

Type: `Void`

**Usage**

This method is not applicable when merging with existing objects—if `setOwnerId` is specified, the `ownerId` field is not overwritten in an existing account or contact.

**setRelatedPersonAccountId (relatedPersonAccountId)**

Sets the ID of the existing person account into which to convert the lead. This value is required only when updating an existing person account.
Signature

```
public Void setRelatedPersonAccountId(ID relatedPersonAccountId)
```

Parameters

```
relatedPersonAccountId
  Type: ID
```

Return Value

Type: Void

```
setRelatedPersonAccountRecord(Entity relatedPersonAccountRecord)
```

Sets the entity record of the new person account into which to convert the lead.

Signature

```
public Void setRelatedPersonAccountRecord(Entity relatedPersonAccountRecord)
```

Parameters

```
relatedPersonAccountRecord
  Type: ID
```

Return Value

Type: Void

```
setSendNotificationEmail(Boolean sendEmail)
```

Specifies whether to send a notification email to the owner specified by `setOwnerId`. The default value is `false`, that is, to not send email.

Signature

```
public Void setSendNotificationEmail(Boolean sendEmail)
```

Parameters

```
sendEmail
  Type: Boolean
```

Return Value

Type: Void

```
LeadConvertResult Class
```

The result of a lead conversion.
Namespace

Database

Usage

An array of LeadConvertResult objects is returned with the `convertLead` Database method. Each element in the LeadConvertResult array corresponds to the sObject array passed as the `SObject[]` parameter in the `convertLead` Database method, that is, the first element in the LeadConvertResult array matches the first element passed in the SObject array, the second element corresponds to the second element, and so on. If only one sObject is passed in, the LeadConvertResult array contains a single element.

LeadConvertResult Methods

The following are methods for `LeadConvertResult`. All are instance methods.

IN THIS SECTION:
  - `getAccountId()`
  - `getContactId()`
  - `getErrors()`
  - `getLeadId()`
  - `getOpportunityId()`
  - `getRelatedPersonAccountId()`
  - `isSuccess()`

`getAccountId()`

The ID of the new account (if a new account was specified) or the ID of the account specified when `convertLead` was invoked.

Signature

public ID getAccountId()

Return Value

Type: ID

`getContactId()`

The ID of the new contact (if a new contact was specified) or the ID of the contact specified when `convertLead` was invoked.

`getErrors()`

If an error occurred, an array of one or more database error objects providing the error code and description.

`getLeadId()`

The ID of the converted lead.

`getOpportunityId()`

The ID of the new opportunity, if one was created when `convertLead` was invoked.

`getRelatedPersonAccountId()`

The ID of the new or existing person account specified when `convertLead` was invoked.

`isSuccess()`

A Boolean value that is set to `true` if the DML operation was successful for this object, `false` otherwise.

2078
Signature
public ID getContactId()

Return Value
Type: ID

getErrors()
If an error occurred, an array of one or more database error objects providing the error code and description.

Signature
public Database.Error[] getErrors()

Return Value
Type: Database.Error[]

getLeadId()
The ID of the converted lead.

Signature
public ID getLeadId()

Return Value
Type: ID

getOpportunityId()
The ID of the new opportunity, if one was created when convertLead was invoked.

Signature
public ID getOpportunityId()

Return Value
Type: ID

getRelatedPersonAccountId()
The ID of the new or existing person account specified when convertLead was invoked.

Signature
public ID getRelatedPersonAccountId()
Return Value
Type: ID

**isSuccess()**
A Boolean value that is set to `true` if the DML operation was successful for this object, `false` otherwise.

Signature

```java
public Boolean isSuccess()
```

Return Value
Type: Boolean

**MergeResult Class**
Contains the result of a merge Database method operation.

**Namespace**
`Database`

**Usage**
The `Database.merge` method returns a `Database.MergeResult` object for each merged record.

**MergeResult Methods**
The following are methods for `MergeResult`. All are instance methods.

**IN THIS SECTION:**
- `getErrors()`
  Returns a list of `Database.Error` objects representing the errors encountered, if any, during a merge operation using the `Database.merge` method. If no error occurred, returns null.
- `getId()`
  Returns the ID of the master record into which other records were merged.
- `getMergedRecordIds()`
  Returns the IDs of the records merged into the master record.
- `getUpdatedRelatedIds()`
  Returns the IDs of all related records that were reparented as a result of the merge that are viewable by the user sending the merge call.
- `isSuccess()`
  Indicates whether the merge was successful (`true`) or not (`false`).
getErrors()
Returns a list of Database.Error objects representing the errors encountered, if any, during a merge operation using the Database.merge method. If no error occurred, returns null.

Signature
public List<Database.Error> getErrors()

Return Value
Type: List<Database.Error>

get_id()
Returns the ID of the master record into which other records were merged.

Signature
public Id getId()

Return Value
Type: ID

getMergedRecordIds()
Returns the IDs of the records merged into the master record.

Signature
public List<String> getMergedRecordIds()

Return Value
Type: List<String>

getUpdatedRelatedIds()
Returns the IDs of all related records that were reparented as a result of the merge that are viewable by the user sending the merge call.

Signature
public List<String> getUpdatedRelatedIds()

Return Value
Type: List<String>
**isSuccess()**
Indicates whether the merge was successful (`true`) or not (`false`).

**Signature**
```
public Boolean isSuccess()
```

**Return Value**
Type: `Boolean`

---

**QueryLocator Class**
Represents the record set returned by `Database.getQueryLocator` and used with Batch Apex.

**Namespace**
`Database`

**QueryLocator Methods**
The following are methods for `QueryLocator`. All are instance methods.

**IN THIS SECTION:**
- `getQuery()`
  Returns the query used to instantiate the `Database.QueryLocator` object. This is useful when testing the `start` method.
- `iterator()`
  Returns a new instance of a query locator iterator.

**getQuery()**
Returns the query used to instantiate the `Database.QueryLocator` object. This is useful when testing the `start` method.

**Signature**
```
public String getQuery()
```

**Return Value**
Type: `String`

**Usage**
You can’t use the `FOR UPDATE` keywords with a `getQueryLocator` query to lock a set of records. The set of records in the batch is determined when the `start` method is run.
iterator()

Returns a new instance of a query locator iterator.

Signature

public Database.QueryLocatorIterator iterator()

Return Value

Type: Database.QueryLocatorIterator

Usage

⚠️ Warning: To iterate over a query locator, save the iterator instance that this method returns in a variable and then use this variable to iterate over the collection. Calling iterator every time you want to perform an iteration can result in incorrect behavior because each call returns a new iterator instance.

For an example, see QueryLocatorIterator Class.

QueryLocatorIterator Class

Represents an iterator over a query locator record set.

Namespace

Database

Example

This sample shows how to obtain an iterator for a query locator, which contains five accounts. This sample calls hasNext and next to get each record in the collection.

```java
// Get a query locator
Database.QueryLocator q = Database.getQueryLocator(
    [SELECT Name FROM Account LIMIT 5]);
// Get an iterator
Database.QueryLocatorIterator it = q.iterator();

// Iterate over the records
while (it.hasNext())
{
    Account a = (Account)it.next();
}
```
QueryLocatorIterator Methods

The following are methods for QueryLocatorIterator. All are instance methods.

**IN THIS SECTION:**

**hasNext()**
Returns `true` if there are one or more records remaining in the collection; otherwise, returns `false`.

**next()**
Advances the iterator to the next sObject record and returns the sObject.

**hasNext()**
Returns `true` if there are one or more records remaining in the collection; otherwise, returns `false`.

**Signature**
```java
public Boolean hasNext()
```

**Return Value**
Type: Boolean

**next()**
Advances the iterator to the next sObject record and returns the sObject.

**Signature**
```java
public sObject next()
```

**Return Value**
Type: sObject

**Usage**
Because the return value is the generic sObject type, you must cast it if using a more specific type. For example:

```java
Account a = (Account)myIterator.next();
```

**Example**
```java
Account a = (Account)myIterator.next();
```
SaveResult Class

The result of an insert or update DML operation returned by a Database method.

Namespace

Database

Usage

An array of SaveResult objects is returned with the insert and update database methods. Each element in the SaveResult array corresponds to the sObject array passed as the sObject[] parameter in the Database method, that is, the first element in the SaveResult array matches the first element passed in the sObject array, the second element corresponds with the second element, and so on. If only one sObject is passed in, the SaveResult array contains a single element.

A SaveResult object is generated when a new or existing Salesforce record is saved.

Example

The following example shows how to obtain and iterate through the returned Database.SaveResult objects. It inserts two accounts using Database.insert with a false second parameter to allow partial processing of records on failure. One of the accounts is missing the Name required field, which causes a failure. Next, it iterates through the results to determine whether the operation was successful or not for each record. It writes the ID of every record that was processed successfully to the debug log, or error messages and fields of the failed records. This example generates one successful operation and one failure.

```java
// Create two accounts, one of which is missing a required field
Account[] accts = new List<Account>{
    new Account(Name='Account1'),
    new Account();
};
Database.SaveResult[] srList = Database.insert(accts, false);

// Iterate through each returned result
for (Database.SaveResult sr : srList) {
    if (sr.isSuccess()) {
        // Operation was successful, so get the ID of the record that was processed
        System.debug('Successfully inserted account. Account ID: ' + sr.getId());
    } else {
        // Operation failed, so get all errors
        for(Database.Error err : sr.getErrors()) {
            System.debug('The following error has occurred: ');
            System.debug(err.getStatusCode() + ': ' + err.getMessage());
            System.debug('Account fields that affected this error: ' + err.getFields());
        }
    }
}
```

SEE ALSO:

Error Class

DuplicateError Class
SaveResult Methods

The following are methods for SaveResult. All are instance methods.

IN THIS SECTION:
  getErrors()  
  If an error occurred, returns an array of one or more database error objects providing the error code and description. If no error occurred, returns an empty set.
  getId()  
  Returns the ID of the sObject you were trying to insert or update.
  isSuccess()  
  Returns a Boolean that is set to true if the DML operation was successful for this object, false otherwise.

getErrors()

If an error occurred, returns an array of one or more database error objects providing the error code and description. If no error occurred, returns an empty set.

Signature

public Database.Error[] getErrors()

Return Value

Type: Database.Error[]

getId()

Returns the ID of the sObject you were trying to insert or update.

Signature

public ID getId()

Return Value

Type: ID

Versioned Behavior Changes

Prior to API version 53.0, the method returned a null value if the record wasn’t updated successfully. In API version 53.0 and later, the method returns the sObject ID, regardless of whether the update operation is successful or not.

isSuccess()

Returns a Boolean that is set to true if the DML operation was successful for this object, false otherwise.
Signature

public Boolean isSuccess()

Return Value

Type: Boolean

Example

This example shows the code used to process duplicate records, which are detected when there is an unsuccessful save due to an error. This code is part of a custom application that implements duplicate management when users add a contact. See DuplicateResult Class on page 2092 to check out the entire sample application.

```java
if (!saveResult.isSuccess()) { ... }
```

UndeleteResult Class

The result of an undelete DML operation returned by the Database.undelete method.

Namespace

Database

Usage

An array of Database.UndeleteResult objects is returned with the undelete database method. Each element in the UndeleteResult array corresponds to the sObject array passed as the sObject[] parameter in the undelete Database method; that is, the first element in the UndeleteResult array matches the first element passed in the sObject array, the second element corresponds with the second element, and so on. If only one sObject is passed in, the UndeleteResults array contains a single element.

UndeleteResult Methods

The following are methods for UndeleteResult. All are instance methods.

IN THIS SECTION:

getErrors()
If an error occurred, returns an array of one or more database error objects providing the error code and description. If no error occurred, returns null.

getId()
Returns the ID of the sObject you were trying to undelete.

isSuccess()
Returns a Boolean value that is set to true if the DML operation was successful for this object, false otherwise.

getErrors()
If an error occurred, returns an array of one or more database error objects providing the error code and description. If no error occurred, returns null.
public Database.Error[] getErrors()

Return Value
Type: Database.Error[]

getId()
Returns the ID of the sObject you were trying to undelete.

public ID getId()

Return Value
Type: ID

Usage
If this field contains a value, the object was successfully undeleted. If this field is empty, the operation was not successful for that object.

isSuccess()
Returns a Boolean value that is set to true if the DML operation was successful for this object, false otherwise.

public Boolean isSuccess()

Return Value
Type: Boolean

UpsertResult Class
The result of an upsert DML operation returned by the Database.upsert method.

Namespace
Database

Usage
An array of Database.UpsertResult objects is returned with the upsert database method. Each element in the UpsertResult array corresponds to the sObject array passed as the sObject[] parameter in the upsert Database method; that is, the first element in the UpsertResult array matches the first element passed in the sObject array, the second element corresponds with the second element, and so on. If only one sObject is passed in, the UpsertResults array contains a single element.
**UpsertResult Methods**

The following are methods for `UpsertResult`. All are instance methods.

**IN THIS SECTION:**

- `getErrors()`: If an error occurred, returns an array of one or more database error objects providing the error code and description. If no error occurred, returns an empty set.
- `getId()`: Returns the ID of the sObject you were trying to update or insert.
- `isCreated()`: A Boolean value that is set to `true` if the record was created, `false` if the record was updated.
- `isSuccess()`: Returns a Boolean value that is set to `true` if the DML operation was successful for this object, `false` otherwise.

### getErrors()

If an error occurred, returns an array of one or more database error objects providing the error code and description. If no error occurred, returns an empty set.

**Signature**

```java
public Database.Error[] getErrors()
```

**Return Value**

Type: `Database.Error[]`

### getId()

Returns the ID of the sObject you were trying to update or insert.

**Signature**

```java
public ID getId()
```

**Return Value**

Type: `ID`

**Versioned Behavior Changes**

Prior to API version 53.0, the method returned a null value if the record wasn’t upserted successfully. In API version 53.0 and later, the method returns the sObject ID regardless of whether the update operation is successful.

### isCreated()

A Boolean value that is set to `true` if the record was created, `false` if the record was updated.
Datacloud Namespace

The Datacloud namespace provides classes and methods for retrieving information about duplicate rules. Duplicate rules let you control whether and when users can save duplicate records within Salesforce.

The Datacloud namespace is related to the Duplicate Management feature. For more information, see Manage Duplicate Records in Salesforce Help and Duplicate Management in Trailhead. The Datacloud namespace isn’t related to the Salesforce Data Cloud product. See Data Cloud.

The following are the classes in the Datacloud namespace.

IN THIS SECTION:

  AdditionalInformationMap Class
  Represents other information, if any, about matched records.

  DuplicateResult Class
  Represents the details of a duplicate rule that detected duplicate records and information about those duplicate records.

  FieldDiff Class
  Represents the name of a matching rule field and how the values of the field compare for the duplicate and its matching record.

  FindDuplicates Class
  Performs rule-based searches for duplicate records. The input is an array of sObjects. Each sObject represents a record you want to find duplicates of. The output identifies the detected duplicates for each input sObject based on active duplicate rules for the given object.

  FindDuplicatesByIds Class
  Performs rule-based searches for duplicate records. The input is an array of IDs. Each ID specifies records to search for duplicates among. The duplicates are detected based on the active duplicate rules applicable to the object type corresponding to the input IDs.

Signature

public Boolean isCreated()

Return Value

Type: Boolean

isSuccess()

Returns a Boolean value that is set to true if the DML operation was successful for this object, false otherwise.

Signature

public Boolean isSuccess()

Return Value

Type: Boolean
FindDuplicatesResult Class
Output for rule-based searches for duplicate records. FindDuplicatesResult contains results of detecting duplicates using instances of FindDuplicates or FindDuplicatesByIds classes.

MatchRecord Class
Represents a duplicate record detected by a matching rule.

MatchResult Class
Represents the duplicate results for a matching rule.

AdditionalInformationMap Class
Represents other information, if any, about matched records.

Namespace
Datacloud

IN THIS SECTION:
AdditionalInformationMap Methods

AdditionalInformationMap Methods
The following are methods for AdditionalInformationMap.

IN THIS SECTION:
getName()
Returns the element name.

getValue()
Returns the value of the element.

**getName()**
Returns the element name.

**Signature**
public String getName()

**Return Value**
Type: String

**getValue()**
Returns the value of the element.
Signature

```java
public String getValue()
```

Return Value

Type: String

**DuplicateResult Class**

Represents the details of a duplicate rule that detected duplicate records and information about those duplicate records.

**Namespace**

Datacloud

**Usage**

The `DuplicateResult` class and its methods are available to organizations that use duplicate rules. `DuplicateResult` is contained within `DuplicateError`, which is part of `SaveResult`. `SaveResult` is generated when a user attempts to save a record in Salesforce.

**Example**

This example shows a custom application that lets users add a contact. When a contact is saved, an alert displays if there are duplicate records.

The sample application consists of a Visualforce page and an Apex controller. The Visualforce page is listed first so that you can see how the page makes use of the Apex controller. Save the Apex class first before saving the Visualforce page.

```
<apex:page controller="ContactDedupeController">
    <apex:form>
        <apex:pageBlock title="Duplicate Records" rendered="{!hasDuplicateResult}"
            <apex:pageMessages />
        <apex:pageBlockTable value="{!duplicateRecords}" var="item">
            <apex:column>
                <apex:facet name="header">Name</apex:facet>
                <apex:outputLink value="/ {!item['Id']}">{!item['Name']}</apex:outputLink>
            </apex:column>
            <apex:column>
                <apex:facet name="header">Owner</apex:facet>
                <apex:outputField value="{!item['OwnerId']}"/>
            </apex:column>
            <apex:column>
                <apex:facet name="header">Last Modified Date</apex:facet>
                <apex:outputField value="{!item['LastModifiedDate']}"/>
            </apex:column>
        </apex:pageBlockTable>
    </apex:pageBlock>
    <apex:pageBlock title="Contact" mode="edit">
```

2092
This sample is the Apex controller for the page. This controller contains the action method for the Save button. The `save` method inserts the new contact. If errors are returned, this method iterates through each error, checks if it's a duplicate error, adds the error message to the page, and returns information about the duplicate records to be displayed on the page.

```apex
public class ContactDedupeController {

    // Initialize a variable to hold the contact record you're processing
    private final Contact contact;

    // Initialize a list to hold any duplicate records
    private List<sObject> duplicateRecords;

    // Define variable that’s true if there are duplicate records
    public boolean hasDuplicateResult{get;set;}

    // Define the constructor
    public ContactDedupeController() {

        // Define the values for the contact you're processing based on its ID
        Id id = ApexPages.currentPage().getParameters().get('id');
        this.contact = (id == null) ? new Contact() :
            [SELECT Id, FirstName, LastName, Email, Phone, AccountId
             FROM Contact WHERE Id = :id];

        // Initialize empty list of potential duplicate records
        this.duplicateRecords = new List<sObject>();
        this.hasDuplicateResult = false;
    }

    // Return contact and its values to the Visualforce page for display
    public Contact getContact() {
        return this.contact;
    }

    // Return duplicate records to the Visualforce page for display
    public List<sObject> getDuplicateRecords() {
        return this.duplicateRecords;
    }

    // Process the saved record and handle any duplicates
```
public PageReference save() {

    // Optionally, set DML options here, use “DML” instead of “false”
    // in the insert()
    Database.DMLOptions dml = new Database.DMLOptions();
    dml.DuplicateRuleHeader.allowSave = true;
    dml.DuplicateRuleHeader.runAsCurrentUser = true;
    Database.SaveResult saveResult = Database.insert(contact, false);

    if (!saveResult.isSuccess()) {
        for (Database.Error error : saveResult.getErrors()) {
            // If there are duplicates, an error occurs
            // Process only duplicates and not other errors
            // (e.g., validation errors)
            if (error instanceof Database.DuplicateError) {
                // Handle the duplicate error by first casting it as a
                // DuplicateError class
                // This lets you use methods of that class
                // (e.g., getDuplicateResult())
                Database.DuplicateError duplicateError =
                    (Database.DuplicateError)error;
                Datacloud.DuplicateResult duplicateResult =
                    duplicateError.getDuplicateResult();

                // Display duplicate error message as defined in the duplicate rule
                ApexPages.Message errorMessage = new ApexPages.Message{
                    ApexPages.Severity.ERROR, 'Duplicate Error: ' +
                    duplicateResult.getErrorMessage()};
                ApexPages.addMessage(errorMessage);

                // Get duplicate records
                this.duplicateRecords = new List<sObject>();

                // Return only match results of matching rules that
                // find duplicate records
                Datacloud.MatchResult[] matchResults =
                    duplicateResult.getMatchResults();

                // Just grab first match result (which contains the
                // duplicate record found and other match info)
                Datacloud.MatchResult matchResult = matchResults[0];

                Datacloud.MatchRecord[] matchRecords = matchResult.getMatchRecords();

                // Add matched record to the duplicate records variable
                for (Datacloud.MatchRecord matchRecord : matchRecords) {
                    System.debug('MatchRecord: ' + matchRecord.getRecord());
                    this.duplicateRecords.add(matchRecord.getRecord());
                }

                this.hasDuplicateResult = !this.duplicateRecords.isEmpty();
            }
        }
    }
}
// If there's a duplicate record, stay on the page
return null;

// After save, navigate to the view page:
return (new ApexPages.StandardController(contact)).view();

IN THIS SECTION:
DuplicateResult Methods

SEE ALSO:
SaveResult Class
DuplicateError Class

DuplicateResult Methods
The following are methods for DuplicateResult.

IN THIS SECTION:
getDuplicateRule()  
Returns the developer name of the executed duplicate rule that returned duplicate records.
getErrorMessage()  
Returns the error message configured by the administrator to warn users they may be creating duplicate records. This message is associated with a duplicate rule.
getMatchResults()  
Returns the duplicate records and match information.
isAllowSave()  
Indicates whether the duplicate rule will allow a record that's identified as a duplicate to be saved. Set to true if duplicate rule should allow save; otherwise, false.

getDuplicateRule()  
Returns the developer name of the executed duplicate rule that returned duplicate records.

Signature

public String getDuplicateRule()

Return Value

Type: String
**getErrorMessage()**

Returns the error message configured by the administrator to warn users they may be creating duplicate records. This message is associated with a duplicate rule.

**Signature**

```java
public String getErrorMessage()
```

**Return Value**

Type: String

**Example**

This example shows the code used to display the error message when duplicates are found while saving a new contact. This code is part of a custom application that lets users add a contact. When a contact is saved, an alert displays if there are duplicate records. Review the DuplicateResult Class on page 2092 to check out the entire sample application.

```java
ApexPages.Message errorMessage = new ApexPages.Message(
    ApexPages.Severity.ERROR, 'Duplicate Error: ' +
    duplicateResult.getErrorMessage());
ApexPages.addMessage(errorMessage);
```

**getMatchResults()**

Returns the duplicate records and match information.

**Signature**

```java
public List<Datacloud.MatchResult> getMatchResults()
```

**Return Value**

Type: List<Datacloud.MatchResult>

**Example**

This example shows the code used to return duplicate record and match information and assign it to the `matchResults` variable. This code is part of a custom application that implements duplicate management when users add a contact. See the DuplicateResult Class on page 2092 to check out the entire sample application.

```java
Datacloud.MatchResult[] matchResults =
    duplicateResult.getMatchResults();
```

**isAllowSave()**

Indicates whether the duplicate rule will allow a record that’s identified as a duplicate to be saved. Set to `true` if duplicate rule should allow save; otherwise, `false`.

---

2096
**Signature**

```java
public Boolean isAllowSave()
```

**Return Value**

Type: Boolean

**FieldDiff Class**

Represents the name of a matching rule field and how the values of the field compare for the duplicate and its matching record.

**Namespace**

Datacloud

**IN THIS SECTION:**

FieldDiff Methods

**FieldDiff Methods**

The following are methods for FieldDiff.

**IN THIS SECTION:**

- `getDifference()`
  Returns how the field values compare for the duplicate and its matching record.

- `getName()`
  Returns the name of a field on a matching rule that detected duplicates.

**getDifference()**

Returns how the field values compare for the duplicate and its matching record.

**Signature**

```java
public String getDifference()
```

**Return Value**

Type: String

Possible values include:

- **SAME**: Indicates the field values match exactly.
- **DIFFERENT**: Indicates that the field values do not match.
- **NULL**: Indicates that the field values are a match because both values are blank.
getName()

Returns the name of a field on a matching rule that detected duplicates.

Signature

```java
public String getName()
```

Return Value

Type: String

FindDuplicates Class

Performs rule-based searches for duplicate records. The input is an array of sObjects. Each sObject represents a record you want to find duplicates of. The output identifies the detected duplicates for each input sObject based on active duplicate rules for the given object.

Namespace

Datacloud

IN THIS SECTION:

FindDuplicates Methods

FindDuplicates Methods

The following are methods for FindDuplicates.

IN THIS SECTION:

findDuplicates(sObjects)

Identifies duplicates for sObjects provided and returns a list of FindDuplicatesResult objects.

findDuplicates(sObjects)

Identifies duplicates for sObjects provided and returns a list of FindDuplicatesResult objects.

Usage

Use FindDuplicates to apply active duplicate rules associated with an object to records represented by input sObjects. FindDuplicates uses the duplicate rules for the object that has the same type as the input sObjects.

Input

- All sObjects in the input array must be of the same object type, and that type must correspond to an object type that supports duplicate rules.
- The input array is limited to 50 elements. If you exceed this limit, an exception is thrown with the following message: Configuration error: The number of records to check is greater than the permitted batch size.
Output

- The output of FindDuplicates is an array of objects with the same number of elements as the input array, and in the same order. The output objects encapsulate record IDs for duplicate records. The output objects also contain values from the duplicate records.
- Each element contains an array of DuplicateResult objects. If FindDuplicates doesn’t find any duplicates, the duplicateRule field in DuplicateResult contains the name of the duplicate rule that FindDuplicates applied, but the matchResults array is empty.

Example

```java
Account acct = new Account();
acct.Name = 'Acme';
acct.BillingStreet = '123 Fake St';
acct.BillingCity = 'Springfield';
acct.BillingState = 'VT';
acct.BillingCountry = 'US';

List<Account> acctList = new List<Account>();
acctList.add(acct);

if (Datacloud.FindDuplicates.findDuplicates(acctList).size() == 0) {
    // If the new account doesn't have duplicates, insert it.
    insert(acct);
}
```

Signature

```java
public static List<Datacloud.FindDuplicatesResult> findDuplicates(List<SObject> sObjects)
```

Parameters

- `sObjects` Type: List<SObject>
  
An array of sObjects for which you want to find duplicates.

Return Value

Type: List<FindDuplicatesResult>

FindDuplicatesByIds Class

Performs rule-based searches for duplicate records. The input is an array of IDs. Each ID specifies records to search for duplicates among. The duplicates are detected based on the active duplicate rules applicable to the object type corresponding to the input IDs.

Namespace

Datacloud
IN THIS SECTION:

**FindDuplicatesByIds Methods**

**FindDuplicatesByIds Methods**
The following are methods for `FindDuplicatesByIds`.

**IN THIS SECTION:**

- `findDuplicatesByIds(ids)`
  - Identifies duplicates of sObjects provided and returns a list of `FindDuplicatesResult` objects.

## findDuplicatesByIds(ids)

Identifies duplicates of sObjects provided and returns a list of `FindDuplicatesResult` objects.

### Usage

Use `FindDuplicatesByIds` to apply active duplicate rules associated with an object to records represented by the record IDs. `FindDuplicatesByIds` uses the duplicate rules for the object that has the same type as the input record IDs. For example, if the record ID represents an Account, `FindDuplicatesByIds` uses the duplicate rules associated with the Account object.

### Input

- All record IDs in the input array must be of the same object type, and that type must correspond to an object type that supports duplicate rules.
- The input array is limited to 50 elements. If you exceed this limit, an exception is thrown with the following message: **Configuration error: The number of records to check is greater than the permitted batch size.**

### Output

- The output of `FindDuplicatesByIds` is an array of objects with the same number of elements as the input array, and in the same order. The output objects encapsulate record IDs for duplicate records. The output objects also contain values from the duplicate records.
- Each element contains an array of `DuplicateResult` objects. If `FindDuplicatesByIds` doesn’t find any duplicates, the `duplicateRule` field in `DuplicateResult` contains the name of the duplicate rule that `FindDuplicatesByIds` applied, but the `matchResults` array is empty.

### Example

```apex
Account acct = new Account(name='Salesforce');
insert acct;

List<Id> idList = new List<Id>{};
idList.add(acct.id);

if (Datacloud.FindDuplicatesByIds.findDuplicatesByIds(idList).size() > 0) {
    System.debug('Found duplicates');
}
```
public static List<Datacloud.FindDuplicatesResult> findDuplicatesByIds(List<Id> ids)

Parameters
ids
Type: List<Id>
A list of IDs for which you want to find duplicates.

Return Value
Type: List<Datacloud.FindDuplicatesResult>

FindDuplicatesResult Class
Output for rule-based searches for duplicate records. FindDuplicatesResult contains results of detecting duplicates using instances of FindDuplicates or FindDuplicatesByIds classes.

Namespace
Datacloud

IN THIS SECTION:
FindDuplicatesResult Properties
FindDuplicatesResult Methods

FindDuplicatesResult Properties
The following are properties for FindDuplicatesResult.

IN THIS SECTION:
duplicateresults
A list of DuplicateResult objects representing the results of calling FindDuplicates.findDuplicates(sObjects) or FindDuplicatesByIds.findDuplicatesByIds(ids). Elements in the list correspond to sObjects or IDs in the input list.
errors
A list of Database.Error objects holding errors resulting from calling FindDuplicates.findDuplicates(sObjects) or FindDuplicatesByIds.findDuplicatesByIds(ids).
success
Boolean signifying whether the call to FindDuplicates.findDuplicates(sObjects) or FindDuplicatesByIds.findDuplicatesByIds(ids) was successful.
**duplicateresults**
A list of DuplicateResult objects representing the results of calling `FindDuplicates.findDuplicates(sObjects)` or `FindDuplicatesByIds.findDuplicatesByIds(ids)`. Elements in the list correspond to sObjects or IDs in the input list.

**Signature**
```
public List<Datacloud.DuplicateResult> duplicateresults
```

**Property Value**
Type: `List<DuplicateResult>`

**errors**
A list of Database.Error objects holding errors resulting from calling `FindDuplicates.findDuplicates(sObjects)` or `FindDuplicatesByIds.findDuplicatesByIds(ids)`.

**Signature**
```
public List<Database.Error> errors {get; set;}
```

**Property Value**
Type: `List<Database.Error>`

**success**
Boolean signifying whether the call to `FindDuplicates.findDuplicates(sObjects)` or `FindDuplicatesByIds.findDuplicatesByIds(ids)` was successful.

**Signature**
```
public Boolean success {get; set;}
```

**Property Value**
Type: `Boolean`

**FindDuplicatesResult Methods**
The following are methods for `FindDuplicatesResult`.

**IN THIS SECTION:**
- `getDuplicateResults()`
  Returns a list of DuplicateResult objects representing the results of calling `FindDuplicates.findDuplicates(sObjects)` or `FindDuplicatesByIds.findDuplicatesByIds(ids)`. Elements in the list correspond to sObjects or IDs in the input list.
getErrors()
Returns a list of DatabaseError objects containing errors resulting from calling
FindDuplicates.findDuplicates(sObjects) or FindDuplicatesByIds.findDuplicatesByIds(ids),
if errors were encountered.

isSuccess()
Returns a Boolean signifying whether the call to FindDuplicates.findDuplicates(sObjects) or
FindDuplicatesByIds.findDuplicatesByIds(ids) was successful.

getDuplicateResults()
Returns a list of DuplicateResult objects representing the results of calling
FindDuplicates.findDuplicates(sObjects) or FindDuplicatesByIds.findDuplicatesByIds(ids).
Elements in the list correspond to sObjects or IDs in the input list.

Example

```java
Account acct = new Account(name='Salesforce');
List<Account> acctList = new List<Account>();
acctList.add(acct);

Datacloud.FindDuplicatesResult[] results = Datacloud.FindDuplicates.findDuplicates(acctList);
for (Datacloud.FindDuplicatesResult findDupeResult : results) {
    for (Datacloud.DuplicateResult dupeResult : findDupeResult.getDuplicateResults()) {
        for (Datacloud.MatchResult matchResult : dupeResult.getMatchResults()) {
            for (Datacloud.MatchRecord matchRecord : matchResult.getMatchRecords()) {
                System.debug('Duplicate Record: ' + matchRecord.getRecord());
            }
        }
    }
}
```

Signature

```java
public List<Datacloud.DuplicateResult> getDuplicateResults()
```

Return Value

Type: List<DuplicateResult>

getErrors()

Returns a list of DatabaseError objects containing errors resulting from calling
FindDuplicates.findDuplicates(sObjects) or FindDuplicatesByIds.findDuplicatesByIds(ids),
if errors were encountered.

Signature

```java
public List<Database.Error> getErrors()
```
Return Value
Type: List<Database.Error>

**isSuccess()**

Returns a Boolean signifying whether the call to `FindDuplicates.findDuplicates(sObjects)` or `FindDuplicatesByIds.findDuplicatesByIds(ids)` was successful.

**Signature**

```
public Boolean isSuccess()
```

Return Value
Type: Boolean

**MatchRecord Class**

Represents a duplicate record detected by a matching rule.

**Namespace**

Datacloud

**IN THIS SECTION:**

MatchRecord Methods

**MatchRecord Methods**

The following are methods for MatchRecord.

**IN THIS SECTION:**

- `getAdditionalInformation()`
  Returns other information about a matched record. For example, a `matchGrade` represents the quality of the data for the D&B fields in the matched record.

- `getFieldDiffs()`
  Returns all matching rule fields and how each field value compares for the duplicate and its matching record.

- `getMatchConfidence()`
  Returns the ranking of how similar a matched record’s data is to the data in your request. Must be equal to or greater than the value of the `minMatchConfidence` specified in your request. Returns -1 if unused.

- `getRecord()`
  Returns the fields and field values for the duplicate.
getAdditionalInformation()
Returns other information about a matched record. For example, a matchGrade represents the quality of the data for the D&B fields in the matched record.

Signature
public List<Datacloud.AdditionalInformationMap> getAdditionalInformation()

Return Value
Type: List<Datacloud.AdditionalInformationMap>

getFieldDiffs()
Returns all matching rule fields and how each field value compares for the duplicate and its matching record.

Signature
public List<Datacloud.FieldDiff> getFieldDiffs()

Return Value
Type: List<Datacloud.FieldDiff>

getMatchConfidence()
Returns the ranking of how similar a matched record’s data is to the data in your request. Must be equal to or greater than the value of the minMatchConfidence specified in your request. Returns -1 if unused.

Signature
public Double getMatchConfidence()

Return Value
Type: Double

getRecord()
Returns the fields and field values for the duplicate.

Signature
public SObject getRecord()

Return Value
Type: SObject
**MatchResult Class**

Represents the duplicate results for a matching rule.

**Namespace**

Datacloud

**IN THIS SECTION:**

MatchResult Methods

**MatchResult Methods**

The following are methods for MatchResult.

**IN THIS SECTION:**

- `getEntityType()`: Returns the entity type of the matching rule.
- `getErrors()`: Returns errors that occurred during matching for the matching rule.
- `getMatchEngine()`: Returns the match engine for the matching rule.
- `getMatchRecords()`: Returns information about the duplicates for the matching rule.
- `getRule()`: Returns the developer name of the matching rule.
- `getSize()`: Returns the number of duplicates detected by the matching rule.
- `isSuccess()`: Returns `false` if there’s an error with the matching rule, and `true` if the matching rule successfully ran.

**getEntityType()**

Returns the entity type of the matching rule.

**Signature**

```java
public String getEntityType()
```

**Return Value**

Type: `String`
getErrors()
Returns errors that occurred during matching for the matching rule.

Signature
public List<Database.Error> getErrors()

Return Value
Type: List<Database.Error>

getMatchEngine()
Returns the match engine for the matching rule.

Signature
public String getMatchEngine()

Return Value
Type: String

getMatchRecords()
Returns information about the duplicates for the matching rule.

Signature
public List<Datacloud.MatchRecord> getMatchRecords()

Return Value
Type: List<Datacloud.MatchRecord>

getRule()
Returns the developer name of the matching rule.

Signature
public String getRule()

Return Value
Type: String

getSize()
Returns the number of duplicates detected by the matching rule.
Signature

```java
public Integer getSize()
```

Return Value

Type: Integer

**isSuccess()**

Returns `false` if there's an error with the matching rule, and `true` if the matching rule successfully ran.

Signature

```java
public Boolean isSuccess()
```

Return Value

Type: Boolean

## DataSource Namespace

The **DataSource** namespace provides the classes for the Apex Connector Framework. Use the Apex Connector Framework to develop a custom adapter for Salesforce Connect. Then connect your Salesforce organization to any data anywhere via the Salesforce Connect custom adapter.

The following are the classes in the **DataSource** namespace.

**IN THIS SECTION:**

- **AsyncDeleteCallback Class**
  A callback class that the `Database.deleteAsync` method references. Salesforce calls this class after the remote `deleteAsync` operation is completed. This class provides the compensating transaction in the completion context of the delete operation. Extend this class to define the actions to execute after the remote delete operation finishes execution.

- **AsyncSaveCallback Class**
  A callback class that the `Database.insertAsync` or `Database.updateAsync` method references. Salesforce calls this class after the remote operation is completed. This class provides the compensating transaction in the completion context of the insert or update operation. Extend this class to define the actions to execute after the remote insert or update operation finishes execution.

- **AuthenticationCapability Enum**
  Specifies the types of authentication that can be used to access the external system.

- **AuthenticationProtocol Enum**
  Determines what type of credentials are used to authenticate to the external system.

- **Capability Enum**
  Declares which functional operations the external system supports. Also specifies required endpoint settings for the external data source definition.

- **Column Class**
  Describes a column on a `DataSource.Table`. This class extends the `DataSourceUtil` class and inherits its methods.
ColumnSelection Class
Identifies the list of columns to return during a query or search.

Connection Class
Extend this class to enable your Salesforce org to sync the external system’s schema and to handle queries, searches, and write operations (upsert and delete) of the external data. This class extends the DataSourceUtil class and inherits its methods.

ConnectionParams Class
Contains the credentials for authenticating to the external system.

DataSourceUtil Class
Parent class for the DataSource.Provider, DataSource.Connection, DataSource.Table, and DataSource.Column classes.

DataType Enum
Specifies the data types that are supported by the Apex Connector Framework.

DeleteContext Class
An instance of DeleteContext is passed to the deleteRows() method on your Database.Connection class. The class provides context information about the delete request to the implementor of deleteRows().

DeleteResult Class
Represents the result of a delete operation on an sObject record. The result is returned by the DataSource.deleteRows method of the DataSource.Connection class.

Filter Class
Represents a WHERE clause in a SOSL or SOQL query.

FilterType Enum
Referenced by the type property on a DataSource.Filter.

IdentityType Enum
Determines which set of credentials is used to authenticate to the external system.

Order Class
Contains details about how to sort the rows in the result set. Equivalent to an ORDER BY statement in a SOQL query.

OrderDirection Enum
Specifies the direction for sorting rows based on column values.

Provider Class
Extend this base class to create a custom adapter for Salesforce Connect. The class informs Salesforce of the functional and authentication capabilities that are supported by or required to connect to the external system. This class extends the DataSourceUtil class and inherits its methods.

QueryAggregation Enum
Specifies how to aggregate a column in a query.

QueryContext Class
An instance of QueryContext is provided to the query method on your DataSource.Connection class. The instance corresponds to a SOQL request.

QueryUtils Class
Contains helper methods to locally filter, sort, and apply limit and offset clauses to data rows. This helper class is provided for your convenience during early development and tests, but it isn’t supported for use in production environments.

ReadContext Class
Abstract base class for the QueryContext and SearchContext classes.
SearchContext Class
An instance of SearchContext is provided to the search method on your DataSource.Connection class. The instance corresponds to a search or SOSL request.

SearchUtils Class
Helper class for implementing search on a custom adapter for Salesforce Connect.

Table Class
Describes a table on an external system that the Salesforce Connect custom adapter connects to. This class extends the DataSourceUtil class and inherits its methods.

TableResult Class
Contains the results of a search or query.

TableSelection Class
Contains a breakdown of the SOQL or SOSL query. Its properties represent the FROM, ORDER BY, SELECT, and WHERE clauses in the query.

UpsertContext Class
An instance of UpsertContext is passed to the upsertRows() method on your Datasource.Connection class. This class provides context information about the upsert request to the implementor of upsertRows().

UpsertResult Class
Represents the result of an upsert operation on an external object record. The result is returned by the upsertRows method of the DataSource.Connection class.

DataSource Exceptions
The DataSource namespace contains exception classes.

AsyncDeleteCallback Class
A callback class that the Database.deleteAsync method references. Salesforce calls this class after the remote deleteAsync operation is completed. This class provides the compensating transaction in the completion context of the delete operation. Extend this class to define the actions to execute after the remote delete operation finishes execution.

Namespace
DataSource

IN THIS SECTION:
AsyncDeleteCallback Methods

AsyncDeleteCallback Methods
The following are methods for AsyncDeleteCallback.

IN THIS SECTION:
processDelete(deleteResult)
Override this method to define actions that Salesforce executes after a remote Database.deleteAsync operation is completed. For example, based on the results of the remote operation, you can update custom object data or other data that's stored in the Salesforce org.
processDelete(deleteResult)

Override this method to define actions that Salesforce executes after a remote Database.deleteAsync operation is completed. For example, based on the results of the remote operation, you can update custom object data or other data that's stored in the Salesforce org.

Signature

public void processDelete(Database.DeleteResult deleteResult)

Parameters

deleteResult
   Type: Database.DeleteResult
   The result of the asynchronous delete operation.

Return Value

Type: void

AsyncSaveCallback Class

A callback class that the Database.insertAsync or Database.updateAsync method references. Salesforce calls this class after the remote operation is completed. This class provides the compensating transaction in the completion context of the insert or update operation. Extend this class to define the actions to execute after the remote insert or update operation finishes execution.

Namespace

DataSource

IN THIS SECTION:

AsyncSaveCallback Methods

AsyncSaveCallback Methods

The following are methods for AsyncSaveCallback.

IN THIS SECTION:

processSave(saveResult)

Override this method to define actions that Salesforce executes after the remote Database.insertAsync or Database.updateAsync operation is completed. For example, based on the results of the remote operation, you can update custom object data or other data that's stored in the Salesforce org.

processSave (saveResult)

Override this method to define actions that Salesforce executes after the remote Database.insertAsync or Database.updateAsync operation is completed. For example, based on the results of the remote operation, you can update custom object data or other data that's stored in the Salesforce org.
Signature

```java
public void processSave(Database.SaveResult saveResult)
```

Parameters

`saveResult`  
Type: `Database.SaveResult`  
The result of the asynchronous insert or update operation.

Return Value

Type: `void`

**AuthenticationCapability Enum**

Specifies the types of authentication that can be used to access the external system.

**Usage**

The `DataSource.Provider` class returns `DataSource.AuthenticationCapability` enum values. The returned values determine which authentication settings are available on the external data source definition in Salesforce.

If you set up callouts in your `DataSource.Connection` class, you can specify the callout endpoints as named credentials instead of URLs. If you do so for all callouts, return `ANONYMOUS` as the sole entry in the list of data source authentication capabilities. That way, the external data source definition doesn’t require authentication settings. Salesforce manages all authentication for Apex callouts that specify a named credential as the callout endpoint so that your code doesn’t have to.

**Enum Values**

The following are the values of the `DataSource.AuthenticationCapability` enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANONYMOUS</td>
<td>No credentials are required to authenticate to the external system.</td>
</tr>
<tr>
<td>BASIC</td>
<td>A username and password can be used to authenticate to the external system.</td>
</tr>
<tr>
<td>CERTIFICATE</td>
<td>A security certificate can be supplied when establishing each connection to the external system.</td>
</tr>
<tr>
<td>OAUTH</td>
<td>OAuth can be used to authenticate to the external system.</td>
</tr>
</tbody>
</table>

**AuthenticationProtocol Enum**

Determines what type of credentials are used to authenticate to the external system.

**Enum Values**

The following are the values of the `DataSource.AuthenticationProtocol` enum.
Capability Enum

Declares which functional operations the external system supports. Also specifies required endpoint settings for the external data source definition.

Usage

The `DataSource.Provider` class returns `DataSource.Capability` enum values, which:

- Specify the functional capabilities of the external system.
- Determine which endpoint settings are available on the external data source definition in Salesforce.

Enum Values

The following are the values of the `DataSource.Capability` enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>QUERY_PAGINATION_SERVER_DRIVEN</td>
<td>With server-driven paging, the external system determines the page sizes and batch boundaries. The external system’s paging settings can optimize the external system’s performance and improve the load times for external objects in your org. Also, the external data set can change while your users or the Lightning Platform are paging through the result set. Typically, server-driven paging adjusts batch boundaries to accommodate changing data sets more effectively than client-driven paging. If you enable server-driven paging on an external data source, Salesforce ignores the requested page sizes, including the default <code>queryMore()</code> batch size of 500 rows. The pages returned by the external system determine the batches, but each page can’t exceed 2,000 rows. Also, the Apex code must generate a query token and use it to determine and fetch the next batch of results.</td>
</tr>
<tr>
<td>QUERY_TOTAL_SIZE</td>
<td>The external system can provide the total number of rows that meet the query criteria, even when requested to return a smaller batch size. This capability enables you to simplify how you paginate results by using <code>queryMore()</code></td>
</tr>
<tr>
<td>REQUIRE_ENDPOINT</td>
<td>Requires the administrator to specify the endpoint in the URL field in the external data source definition.</td>
</tr>
<tr>
<td>REQUIRE_HTTPS</td>
<td>Requires the endpoint URL to use secure HTTP. If <code>REQUIRE_ENDPOINT</code> isn’t declared, <code>REQUIRE_HTTPS</code> is ignored.</td>
</tr>
<tr>
<td>ROW_CREATE</td>
<td>Allows creating of external data.</td>
</tr>
<tr>
<td>ROW_DELETE</td>
<td>Allows deleting external data.</td>
</tr>
<tr>
<td>Value</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ROW_QUERY</td>
<td>Allows API and SOQL queries of the external data. Also allows reports on the external objects.</td>
</tr>
<tr>
<td>ROW_UPDATE</td>
<td>Allows updating external data.</td>
</tr>
<tr>
<td>SEARCH</td>
<td>Allows SOSL and Salesforce searches of the external data.</td>
</tr>
<tr>
<td></td>
<td>When the custom adapter declares the SEARCH capability, you can control which external objects are searchable by selecting or deselecting Allow Search on each external object. However, syncing always overwrites the external object’s search status to match the search status of the external data source. Only text, text area, and long text area fields on external objects can be searched. If an external object has no searchable fields, searches on that object return no records.</td>
</tr>
</tbody>
</table>

SEE ALSO:
Salesforce Help: Validate and Sync an External Data Source

Column Class

Describes a column on a DataSource.Table. This class extends the DataSourceUtil class and inherits its methods.

Namespace

DataSource

Usage

A list of column metadata is provided by the DataSource.Connection class when the sync() method is invoked. Each column can become a field on an external object.

The metadata is stored in Salesforce. Updating the Apex code to return new or updated values for the column metadata doesn’t automatically update the stored metadata in Salesforce.

IN THIS SECTION:
Column Properties
Column Methods

Column Properties

The following are properties for Column.

IN THIS SECTION:
decimalPlaces
If the data type is numeric, the number of decimal places to the right of the decimal point.
description
Description of what the column represents.

filterable
Whether a result set can be filtered based on the values of the column.

label
User-friendly name for the column that appears in the Salesforce user interface.

length
If the column is a string data type, the number of characters in the column. If the column is a numeric data type, the total number of digits on both sides of the decimal point, but excluding the decimal point.

name
Name of the column in the external system.

referenceTargetField
API name of the custom field on the parent object whose values are compared against this column’s values. Matching values identify related records in an indirect lookup relationship. Applies only when the column’s data type is INDIRECT_LOOKUP_TYPE. For other data types, this value is ignored.

referenceTo
API name of the parent object in the relationship that’s represented by this column. Applies only when the column’s data type is LOOKUP_TYPE, EXTERNAL_LOOKUP_TYPE, or INDIRECT_LOOKUP_TYPE. For other data types, this value is ignored.

sortable
Whether a result set can be sorted based on the values of the column via an ORDER BY clause.

type
Data type of the column.

decimalPlaces
If the data type is numeric, the number of decimal places to the right of the decimal point.

Signature
public Integer decimalPlaces {get; set;}

Property Value
Type: Integer

description
Description of what the column represents.

Signature
public String description {get; set;}

Property Value
Type: String
**filterable**
Whether a result set can be filtered based on the values of the column.

**Signature**
```
public Boolean filterable {get; set;}
```

**Property Value**
Type: Boolean

**label**
User-friendly name for the column that appears in the Salesforce user interface.

**Signature**
```
public String label {get; set;}
```

**Property Value**
Type: String

**length**
If the column is a string data type, the number of characters in the column. If the column is a numeric data type, the total number of digits on both sides of the decimal point, but excluding the decimal point.

**Signature**
```
public Integer length {get; set;}
```

**Property Value**
Type: Integer

**name**
Name of the column in the external system.

**Signature**
```
public String name {get; set;}
```

**Property Value**
Type: String
**referenceTargetField**

API name of the custom field on the parent object whose values are compared against this column’s values. Matching values identify related records in an indirect lookup relationship. Applies only when the column’s data type is `INDIRECT_LOOKUP_TYPE`. For other data types, this value is ignored.

**Signature**

```java
public String referenceTargetField {get; set;}
```

**Property Value**

Type: `String`

**referenceTo**

API name of the parent object in the relationship that’s represented by this column. Applies only when the column’s data type is `LOOKUP_TYPE`, `EXTERNAL_LOOKUP_TYPE`, or `INDIRECT_LOOKUP_TYPE`. For other data types, this value is ignored.

**Signature**

```java
public String referenceTo {get; set;}
```

**Property Value**

Type: `String`

**sortable**

Whether a result set can be sorted based on the values of the column via an `ORDER BY` clause.

**Signature**

```java
public Boolean sortable {get; set;}
```

**Property Value**

Type: `Boolean`

**type**

Data type of the column.

**Signature**

```java
public DataSource.DataType type {get; set;}
```

**Property Value**

Type: `DataSource.DataType`
Column Methods

The following are methods for Column.

IN THIS SECTION:

boolean(name)
Returns a new column of data type BOOLEAN_TYPE.

externalLookup(name, domain)
Returns a new column of data type EXTERNAL_LOOKUP_TYPE.

get(name, label, description, isSortable, isFilterable, type, length, decimalPlaces, referenceTo, referenceTargetField)
Returns a new column with the ten specified Column property values.

get(name, label, description, isSortable, isFilterable, type, length, decimalPlaces)
Returns a new column with the eight specified Column property values.

get(name, label, description, isSortable, isFilterable, type, length)
Returns a new column with the seven specified Column property values.

indirectLookup(name, domain, targetField)
Returns a new column of data type INDIRECT_LOOKUP_TYPE.

integer(name, length)
Returns a new numeric column with no decimal places using the specified name and length.

lookup(name, domain)
Returns a new column of data type LOOKUP_TYPE.

number(name, length, decimalPlaces)
Returns a new column of data type NUMBER_TYPE.

text(name, label, length)
Returns a new column of data type STRING_SHORT_TYPE or STRING_LONG_TYPE, with the specified name, label, and length.

text(name, length)
Returns a new column of data type STRING_SHORT_TYPE or STRING_LONG_TYPE, with the specified name and length.

text(name)
Returns a new column of data type STRING_SHORT_TYPE with the specified name and the length of 255 characters.

textarea(name)
Returns a new column of data type STRING_LONG_TYPE with the specified name and the length of 32,000 characters.

url(name, length)
Returns a new column of data type URL_TYPE with the specified name and length.

url(name)
Returns a new column of data type URL_TYPE with the specified name and the length of 1,000 characters.

boolean(name)
Returns a new column of data type BOOLEAN_TYPE.
Signature

public static DataSource.Column boolean(String name)

Parameters

name
  Type: String
  Name of the column.

Return Value

Type: DataSource.Column

externalLookup(name, domain)

Returns a new column of data type EXTERNAL_LOOKUP_TYPE.

Signature

public static DataSource.Column externalLookup(String name, String domain)

Parameters

name
  Type: String
  Name of the column.

domain
  Type: String
  API name of the parent object in the external lookup relationship.

Return Value

Type: DataSource.Column

The returned column has these property values.

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>name</td>
</tr>
<tr>
<td>label</td>
<td>name</td>
</tr>
<tr>
<td>description</td>
<td>name</td>
</tr>
<tr>
<td>isSortable</td>
<td>true</td>
</tr>
<tr>
<td>isFilterable</td>
<td>true</td>
</tr>
<tr>
<td>type</td>
<td>DataSource.DataType.EXTERNAL_LOOKUP_TYPE</td>
</tr>
<tr>
<td>length</td>
<td>255</td>
</tr>
</tbody>
</table>
get(name, label, description, isSortable, isFilterable, type, length, decimalPlaces, referenceTo, referenceTargetField)

Returns a new column with the ten specified Column property values.

Signature

public static DataSource.Column get(String name, String label, String description, Boolean isSortable, Boolean isFilterable, DataSource.DataType type, Integer length, Integer decimalPlaces, String referenceTo, String referenceTargetField)

Parameters

See Column Properties on page 2114 for information about each parameter.

name
  Type: String

label
  Type: String

description
  Type: String

isSortable
  Type: Boolean

isFilterable
  Type: Boolean

type
  Type: DataSource.DataType

length
  Type: Integer

decimalPlaces
  Type: Integer

referenceTo
  Type: String

referenceTargetField
  Type: String

Return Value

Type: DataSource.Column
get(name, label, description, isSortable, isFilterable, type, length, decimalPlaces)

Returns a new column with the eight specified Column property values.

Signature

public static DataSource.Column get(String name, String label, String description, Boolean isSortable, Boolean isFilterable, DataSource.DataType type, Integer length, Integer decimalPlaces)

Parameters

See Column Properties on page 2114 for information about each parameter.

- name
  Type: String
- label
  Type: String
- description
  Type: String
- isSortable
  Type: Boolean
- isFilterable
  Type: Boolean
- type
  Type: DataSource.DataType
- length
  Type: Integer
- decimalPlaces
  Type: Integer

Return Value

Type: DataSource.Column

get(name, label, description, isSortable, isFilterable, type, length)

Returns a new column with the seven specified Column property values.

Signature

public static DataSource.Column get(String name, String label, String description, Boolean isSortable, Boolean isFilterable, DataSource.DataType type, Integer length)

Parameters

See Column Properties on page 2114 for information about each parameter.
name
  Type: String

label
  Type: String

description
  Type: String

isSortable
  Type: Boolean

isFilterable
  Type: Boolean

type
  Type: DataSource.DataType

length
  Type: Integer

Return Value
  Type: DataSource.Column

indirectLookup(name, domain, targetField)

Returns a new column of data type INDIRECT_LOOKUP_TYPE.

Signature

public static DataSource.Column indirectLookup(String name, String domain, String targetField)

Parameters

name
  Type: String

  Name of the column.

domain
  Type: String

  API name of the parent object in the indirect lookup relationship.

targetField
  Type: String

  API name of the custom field on the parent object whose values are compared against this column’s values. Matching values identify related records in an indirect lookup relationship.

Return Value
  Type: DataSource.Column

The returned column has these property values.
**integer(name, length)**

Returns a new numeric column with no decimal places using the specified name and length.

**Signature**

```java
public static DataSource.Column integer(String name, Integer length)
```

**Parameters**

- **name**
  - Type: `String`
  - The column name.

- **length**
  - Type: `Integer`
  - The column length.

**Return Value**

Type: `DataSource.Column`

**lookup(name, domain)**

Returns a new column of data type `LOOKUP_TYPE`.

**Signature**

```java
public static DataSource.Column lookup(String name, String domain)
```
Parameters

name
Type: String
Name of the column.

domain
Type: String
API name of the parent object in the lookup relationship.

Return Value
Type: DataSource.Column
The returned column has these property values.

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>name</td>
</tr>
<tr>
<td>label</td>
<td>name</td>
</tr>
<tr>
<td>description</td>
<td>name</td>
</tr>
<tr>
<td>isSortable</td>
<td>true</td>
</tr>
<tr>
<td>isFilterable</td>
<td>true</td>
</tr>
<tr>
<td>type</td>
<td>DataSource.DataType.LOOKUP_TYPE</td>
</tr>
<tr>
<td>length</td>
<td>255</td>
</tr>
<tr>
<td>decimalPlaces</td>
<td>0</td>
</tr>
<tr>
<td>referenceTo</td>
<td>domain</td>
</tr>
<tr>
<td>referenceTargetField</td>
<td>null</td>
</tr>
</tbody>
</table>

number(name, length, decimalPlaces)
Returns a new column of data type NUMBER_TYPE.

Signature
public static DataSource.Column number(String name, Integer length, Integer decimalPlaces)

Parameters
See Column Properties on page 2114 for information about each parameter.

name
Type: String

length
Type: Integer
decimalPlaces
  Type: Integer

Return Value
Type: DataSource.Column
The returned column has these property values.

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>name</td>
</tr>
<tr>
<td>label</td>
<td>name</td>
</tr>
<tr>
<td>description</td>
<td>name</td>
</tr>
<tr>
<td>isSortable</td>
<td>true</td>
</tr>
<tr>
<td>isFilterable</td>
<td>true</td>
</tr>
<tr>
<td>type</td>
<td>DataSource.DataType.NUMBER_TYPE</td>
</tr>
<tr>
<td>length</td>
<td>length</td>
</tr>
<tr>
<td>decimalPlaces</td>
<td>decimalPlaces</td>
</tr>
</tbody>
</table>

text(name, label, length)
Returns a new column of data type STRING_SHORT_TYPE or STRING_LONG_TYPE, with the specified name, label, and length.

Signature
public static DataSource.Column text(String name, String label, Integer length)

Parameters
name
  Type: String
  Name of the column.

label
  Type: String
  User-friendly name for the column that appears in the Salesforce user interface.

length
  Type: Integer
  Number of characters allowed in the column.

Return Value
Type: DataSource.Column
The returned column has these property values.
**text(name, length)**

Returns a new column of data type `STRING_SHORT_TYPE` or `STRING_LONG_TYPE`, with the specified name and length.

**Signature**

```java
public static DataSource.Column text(String name, Integer length)
```

**Parameters**

- **name**
  - Type: `String`
  - Name of the column.

- **length**
  - Type: `Integer`
  - Number of characters allowed in the column.

**Return Value**

Type: `DataSource.Column`

The returned column has these property values.

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td><code>name</code></td>
</tr>
<tr>
<td>label</td>
<td><code>label</code></td>
</tr>
<tr>
<td>description</td>
<td><code>label</code></td>
</tr>
<tr>
<td>isSortable</td>
<td>true</td>
</tr>
<tr>
<td>isFilterable</td>
<td>true</td>
</tr>
<tr>
<td>type</td>
<td><code>length</code></td>
</tr>
<tr>
<td>decimalPlaces</td>
<td>0</td>
</tr>
</tbody>
</table>

 DataSource.DataType.STRING_SHORT_TYPE if `length` is 255 or less
 DataSource.DataType.STRING_LONG_TYPE if `length` is greater than 255
**Property** | **Value**
---|---
type | DataSource.DataType.STRING_SHORT_TYPE if length is 255 or less
| DataSource.DataType.STRING_LONG_TYPE if length is greater than 255
length | length
decimalPlaces | 0

**text(name)**

Returns a new column of data type STRING_SHORT_TYPE with the specified name and the length of 255 characters.

**Signature**

```java
public static DataSource.Column text(String name)
```

**Parameters**

- **name**
  - Type: String
  - Name of the column.

**Return Value**

Type: DataSource.Column

The returned column has these property values.

**Property** | **Value**
---|---
name | name
label | name
description | name
isSortable | true
isFilterable | true
type | DataSource.DataType.STRING_SHORT_TYPE
length | 255
decimalPlaces | 0

**textarea(name)**

Returns a new column of data type STRING_LONG_TYPE with the specified name and the length of 32,000 characters.
Signature

public static DataSource.Column textarea(String name)

Parameters

name
Type: String
Name of the column.

Return Value
Type: DataSource.Column
The returned column has these property values.

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>name</td>
</tr>
<tr>
<td>label</td>
<td>name</td>
</tr>
<tr>
<td>description</td>
<td>name</td>
</tr>
<tr>
<td>isSortable</td>
<td>true</td>
</tr>
<tr>
<td>isFilterable</td>
<td>true</td>
</tr>
<tr>
<td>type</td>
<td>DataSource.DataType.STRING_LONG_TYPE</td>
</tr>
<tr>
<td>length</td>
<td>32000</td>
</tr>
<tr>
<td>decimalPlaces</td>
<td>0</td>
</tr>
</tbody>
</table>

url(name, length)
Returns a new column of data type URL_TYPE with the specified name and length.

Signature

public static DataSource.Column url(String name, Integer length)

Parameters

name
Type: String
Name of the column.

length
Type: Integer
Number of characters allowed in the column.
Return Value
Type: DataSource.Column
The returned column has these property values.

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>name</td>
</tr>
<tr>
<td>label</td>
<td>name</td>
</tr>
<tr>
<td>description</td>
<td>name</td>
</tr>
<tr>
<td>isSortable</td>
<td>true</td>
</tr>
<tr>
<td>isFilterable</td>
<td>true</td>
</tr>
<tr>
<td>type</td>
<td>DataSource.DataType.URL_TYPE</td>
</tr>
<tr>
<td>length</td>
<td>length</td>
</tr>
<tr>
<td>decimalPlaces</td>
<td>0</td>
</tr>
</tbody>
</table>

`url(name)`
Returns a new column of data type `URL_TYPE` with the specified name and the length of 1,000 characters.

Signature

```java
public static DataSource.Column url(String name)
```

Parameters

name
Type: String
Name of the column.

Return Value
Type: DataSource.Column
The returned column has these property values.

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>name</td>
</tr>
<tr>
<td>label</td>
<td>name</td>
</tr>
<tr>
<td>description</td>
<td>name</td>
</tr>
<tr>
<td>isSortable</td>
<td>true</td>
</tr>
<tr>
<td>isFilterable</td>
<td>true</td>
</tr>
<tr>
<td>type</td>
<td>DataSource.DataType.URL_TYPE</td>
</tr>
</tbody>
</table>
### ColumnSelection Class

Identifies the list of columns to return during a query or search.

### Namespace

**DataSource Namespace**

### Usage

This class is associated with the `SELECT` clause for a SOQL query, or the `RETURNING` clause for a SOSL query.

### ColumnSelection Properties

The following are properties for `ColumnSelection`.

#### aggregation

How to aggregate the column’s data.

#### columnName

Name of the selected column.

#### tableName

Name of the column’s table.

#### aggregation

How to aggregate the column’s data.

**Signature**

```
public DataSource.QueryAggregation aggregation {get; set;}
```

**Property Value**

Type: `DataSource.QueryAggregation`
**columnName**
Name of the selected column.

**Signature**
```java
public String columnName {get; set;}
```

**Property Value**
Type: String

**tableName**
Name of the column's table.

**Signature**
```java
public String tableName {get; set;}
```

**Property Value**
Type: String

**Connection Class**
Extend this class to enable your Salesforce org to sync the external system's schema and to handle queries, searches, and write operations (upsert and delete) of the external data. This class extends the `DataSourceUtil` class and inherits its methods.

**Namespace**
`DataSource`

**Usage**
Your `DataSource.Connection` and `DataSource.Provider` classes compose a custom adapter for Salesforce Connect. Changing the `sync` method on the `DataSource.Connection` class doesn’t automatically resync any external objects.

**Example**
```java
global class SampleDataSourceConnection extends DataSource.Connection {

global SampleDataSourceConnection(DataSource.ConnectionParams connectionParams) {
}

override global List<DataSource.Table> sync() {
    List<DataSource.Table> tables = new List<DataSource.Table>();
    List<DataSource.Column> columns;
    columns = new List<DataSource.Column>();
    columns.add(DataSource.Column.text('Name', 255));
    columns.add(DataSource.Column.text('ExternalId', 255));
    ```
columns.add(DataSource.Column.url('DisplayUrl'));
tables.add(DataSource.Table.get('Sample', 'Title', columns));
return tables;
}

override global DataSource.TableResult query(DataSource.QueryContext c) {
return DataSource.TableResult.get(c, DataSource.QueryUtils.process(c, getRows()));
}

override global List<DataSource.TableResult> search(DataSource.SearchContext c) {
List<DataSource.TableResult> results = new List<DataSource.TableResult>();
for (DataSource.TableSelection tableSelection : c.tableSelections) {
    results.add(DataSource.TableResult.get(tableSelection, getRows()));
}
return results;

// Helper method to get record values from the external system for the Sample table.
private List<Map<String, Object>> getRows () {
    // Get row field values for the Sample table from the external system via a callout.
    HttpResponse response = makeGetCallout();
    // Parse the JSON response and populate the rows.
    Map<String, Object> m = (Map<String, Object>)JSON.deserializeUntyped(response.getBody());
    Map<String, Object> error = (Map<String, Object>)m.get('error');
    if (error != null) {
        throwException(string.valueOf(error.get('message')));
    }
    List<Map<String, Object>> rows = new List<Map<String, Object>>();
    List<Object> jsonRows = (List<Object>)m.get('value');
    if (jsonRows == null) {
        rows.add(foundRow(m));
    } else {
        for (Object jsonRow : jsonRows) {
            Map<String, Object> row = (Map<String, Object>)jsonRow;
            rows.add(foundRow(row));
        }
    }
    return rows;
}

global override List<DataSource.UpsertResult> upsertRows(DataSource.UpsertContext context) {
if (context.tableSelected == 'Sample') {
    List<DataSource.UpsertResult> results = new List<DataSource.UpsertResult>();
    List<Map<String, Object>> rows = context.rows;
    for (Map<String, Object> row : rows) {
        // Make a callout to insert or update records in the external system.
        HttpResponse response;
        // Determine whether to insert or update a record.
if (row.get('ExternalId') == null) {
    // Send a POST HTTP request to insert new external record.
    // Make an Apex callout and get HttpResponse.
    response = makePostCallout(
        '{"name":"' + row.get('Name') + '","ExternalId":"' + row.get('ExternalId') + '"
    };
}
else {
    // Send a PUT HTTP request to update an existing external record.
    // Make an Apex callout and get HttpResponse.
    response = makePutCallout(
        '{"name":"' + row.get('Name') + '","ExternalId":"' + row.get('ExternalId') + '",
            String.valueOf(row.get('ExternalId')));
    }
    // Check the returned response.
    // First, deserialize it.
    Map<String, Object> m = (Map<String, Object>)JSON.deserializeUntyped(response.getBody());
    if (response.getStatusCode() == 200) {
        results.add(DataSource.UpsertResult.success(String.valueOf(m.get('id'))));
    }
    else {
        results.add(DataSource.UpsertResult.failure( String.valueOf(m.get('id')),
        'The callout resulted in an error: ' + response.getStatusCode()));
    }
}
return results;
return null;
}

global override List<DataSource.DeleteResult> deleteRows(DataSource.DeleteContext context) {
    if (context.tableSelected == 'Sample') {
        List<DataSource.DeleteResult> results = new List<DataSource.DeleteResult>();
        for (String externalId : context.externalIds) {
            HttpResponse response = makeDeleteCallout(externalId);
            if (response.getStatusCode() == 200) {
                results.add(DataSource.DeleteResult.success(externalId));
            }
            else {
                results.add(DataSource.DeleteResult.failure(externalId,
                    'Callout delete error:' + response.getBody()));
            }
        }
        return results;
    }
    return null;
}
Helper methods

// Make a GET callout
private static HttpResponse makeGetCallout() {
    HttpResponse response;
    // Make callout
    // ...
    return response;
}

// Populate a row based on values from the external system.
private Map<String,Object> foundRow(Map<String,Object> foundRow) {
    Map<String,Object> row = new Map<String,Object>();
    row.put('ExternalId', string.valueOf(foundRow.get('Id')));
    row.put('DisplayUrl', string.valueOf(foundRow.get('DisplayUrl')));
    row.put('Name', string.valueOf(foundRow.get('Name')));
    return row;
}

// Make a POST callout
private static HttpResponse makePostCallout(String jsonBody) {
    HttpResponse response;
    // Make callout
    // ...
    return response;
}

// Make a PUT callout
private static HttpResponse makePutCallout(String jsonBody, String externalID) {
    HttpResponse response;
    // Make callout
    // ...
    return response;
}

// Make a DELETE callout
private static HttpResponse makeDeleteCallout(String externalID) {
    HttpResponse response;
    // Make callout
    // ...
    return response;
}

IN THIS SECTION:

Connection Methods

Connection Methods
The following are methods for Connection.
IN THIS SECTION:

**deleteRows(deleteContext)**
Invoked when external object records are deleted via the Salesforce user interface, APIs, or Apex.

**query(queryContext)**
Invoked by a SOQL query of an external object. A SOQL query is generated and executed when a user visits an external object’s list view or record detail page in Salesforce. Returns the results of the query.

**search(searchContext)**
Invoked by a SOSL query of an external object or when a user performs a Salesforce global search that also searches external objects. Returns the results of the query.

**sync()**
Invoked when an administrator clicks Validate and Sync on the external data source detail page. Returns a list of tables that describe the external system’s schema.

**upsertRows(upsertContext)**
Invoked when external object records are created or updated via the Salesforce user interface, APIs, or Apex.

---

**deleteRows (deleteContext)**
Invoked when external object records are deleted via the Salesforce user interface, APIs, or Apex.

**Signature**

```java
public List<DataSource.DeleteResult> deleteRows(DataSource.DeleteContext deleteContext)
```

**Parameters**

`deleteContext`
Type: `DataSource.DeleteContext`
Contains context information about the delete request.

**Return Value**
Type: `List<DataSource.DeleteResult>`
The results of the delete operation.

---

**query (queryContext)**
Invoked by a SOQL query of an external object. A SOQL query is generated and executed when a user visits an external object’s list view or record detail page in Salesforce. Returns the results of the query.

**Signature**

```java
public DataSource.TableResult query(DataSource.QueryContext queryContext)
```

**Parameters**

`queryContext`
Type: `DataSource.QueryContext`
Represents the query to run against a data table.

**Return Value**

Type: `DataSource.TableResult`

**search(searchContext)**

Invoked by a SOSL query of an external object or when a user performs a Salesforce global search that also searches external objects. Returns the results of the query.

**Signature**

```java
public List<DataSource.TableResult> search(DataSource.SearchContext searchContext)
```

**Parameters**

`searchContext`

Type: `DataSource.SearchContext`

Represents the query to run against an external data table.

**Return Value**

Type: `List<DataSource.TableResult>`

**sync()**

Invoked when an administrator clicks **Validate and Sync** on the external data source detail page. Returns a list of tables that describe the external system’s schema.

**Signature**

```java
public List<DataSource.Table> sync()
```

**Return Value**

Type: `List<DataSource.Table>`

Each returned table can be used to create an external object in Salesforce. On the Validate External Data Source page, the administrator views the list of returned tables and selects which tables to sync. When the administrator clicks **Sync**, an external object is created for each selected table. Each column within the selected tables also becomes a field in the external object.

**upsertRows(upsertContext)**

Invoked when external object records are created or updated via the Salesforce user interface, APIs, or Apex.

**Signature**

```java
public List<DataSource.UpsertResult> upsertRows(DataSource.UpsertContext upsertContext)
```
Parameters

_upsertContext_
Type: `DataSource.UpsertContext`
Contains context information about the upsert request.

Return Value

Type: `List<DataSource.UpsertResult>`
The results of the upsert operation.

ConnectionParams Class
Contains the credentials for authenticating to the external system.

Namespace
`DataSource`

Usage

If your extension of the `DataSource.Provider` class returns `DataSource.AuthenticationCapability` values that indicate support for authentication, the `DataSource.Connection` class is instantiated with a `DataSource.ConnectionParams` instance in the constructor.

The authentication credentials in the `DataSource.ConnectionParams` instance depend on the `Identity Type` field of the external data source definition in Salesforce.

- If `Identity Type` is set to `Named Principal`, the credentials come from the external data source definition.
- If `Identity Type` is set to `Per User`:
  - For queries and searches, the credentials are specific to the current user who invokes the query or search. The credentials come from the user’s authentication settings for the external system.
  - For administrative connections, such as syncing the external system’s schema, the credentials come from the external data source definition.

The values in this class can appear in debug logs and can be accessed by users who have the “Author Apex” permission. If you require better security, we recommend that you specify named credentials instead of URLs as your Apex callout endpoints. Salesforce manages all authentication for Apex callouts that specify a named credential as the callout endpoint so that your code doesn’t have to.

IN THIS SECTION:

ConnectionParams Properties

ConnectionParams Properties
The following are properties for `ConnectionParams`. 
In this section:

- **certificateName**
  The name of the certificate for establishing each connection to the external system.

- **endpoint**
  The URL of the external system.

- **oauthToken**
  The OAuth token that’s issued by the external system.

- **password**
  The password for authenticating to the external system.

- **principalType**
  An instance of `DataSource.IdentityType`, which determines which set of credentials to use to access the external system.

- **protocol**
  The type of protocol that’s used to authenticate to the external system.

- **repository**
  Reserved for future use.

- **username**
  The username for authenticating to the external system.

**certificateName**
The name of the certificate for establishing each connection to the external system.

**Signature**

```java
public String certificateName {get; set;}
```

**Property Value**

Type: `String`

The value comes from the external data source definition in Salesforce.

**endpoint**
The URL of the external system.

**Signature**

```java
public String endpoint {get; set;}
```

**Property Value**

Type: `String`

The value comes from the external data source definition in Salesforce.
oauthToken
The OAuth token that's issued by the external system.

Signature
public String oauthToken {get; set;}

Property Value
Type: String

password
The password for authenticating to the external system.

Signature
public String password {get; set;}

Property Value
Type: String
The value depends on the Identity Type field of the external data source definition in Salesforce.

• If Identity Type is set to Named Principal, the credentials come from the external data source definition.

• If Identity Type is set to Per User:
  – For queries and searches, the credentials are specific to the current user who invokes the query or search. The credentials come from the user’s authentication settings for the external system.
  – For administrative connections, such as syncing the external system’s schema, the credentials come from the external data source definition.

principalType
An instance of DataSource.IdentityType, which determines which set of credentials to use to access the external system.

Signature
public DataSource.IdentityType principalType {get; set;}

Property Value
Type: DataSource.IdentityType

protocol
The type of protocol that's used to authenticate to the external system.
**Signature**
```
public DataSource.AuthenticationProtocol protocol {get; set;}
```

**Property Value**
Type: `DataSource.AuthenticationProtocol`

**repository**
Reserved for future use.

**Signature**
```
public String repository {get; set;}
```

**Property Value**
Type: `String`
Reserved for future use.

**username**
The username for authenticating to the external system.

**Signature**
```
public String username {get; set;}
```

**Property Value**
Type: `String`
The value depends on the `Identity Type` field of the external data source definition in Salesforce.

- If `Identity Type` is set to `Named Principal`, the credentials come from the external data source definition.
- If `Identity Type` is set to `Per User`:
  - For queries and searches, the credentials are specific to the current user who invokes the query or search. The credentials come from the user’s authentication settings for the external system.
  - For administrative connections, such as syncing the external system’s schema, the credentials come from the external data source definition.

**DataSourceUtil Class**
Parent class for the `DataSource.Provider`, `DataSource.Connection`, `DataSource.Table`, and `DataSource.Column` classes.

**Namespace**
`DataSource`
DataSourceUtil Methods

The following are methods for.DataSourceUtil.

logWarning(message)
Logs the error message in the debug log.

Signature
public void logWarning(String message)

Parameters
message
Type: String
The error message.

Return Value
Type: void

throwException(message)
Throws a DataSourceException and displays the provided message to the user.

Signature
public void throwException(String message)

Parameters
message
Type: String
Error message to display to the user.

Return Value
Type: void
DataType Enum

Specifies the data types that are supported by the Apex Connector Framework.

Usage

The `DataSource.DataType` enum is referenced by the `type` property on the `DataSource.Column` class.

Enum Values

The following are the values of the `DataSource.DataType` enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOOLEAN_TYPE</td>
<td>Boolean</td>
</tr>
<tr>
<td>DATETIME_TYPE</td>
<td>Date/time</td>
</tr>
<tr>
<td>EXTERNAL_LOOKUP_TYPE</td>
<td>External lookup relationship</td>
</tr>
<tr>
<td>INDIRECT_LOOKUP_TYPE</td>
<td>Indirect lookup relationship</td>
</tr>
<tr>
<td>LOOKUP_TYPE</td>
<td>Lookup relationship</td>
</tr>
<tr>
<td>NUMBER_TYPE</td>
<td>Number</td>
</tr>
<tr>
<td>STRING_LONG_TYPE</td>
<td>Long text area</td>
</tr>
<tr>
<td>STRING_SHORT_TYPE</td>
<td>Text area</td>
</tr>
<tr>
<td>URL_TYPE</td>
<td>URL</td>
</tr>
</tbody>
</table>

DeleteContext Class

An instance of `DeleteContext` is passed to the `deleteRows()` method on your `Database.Connection` class. The class provides context information about the delete request to the implementor of `deleteRows()`.

Namespace

`DataSource`

Usage

The Apex Connector Framework creates context for operations. Context is comprised of parameters about the operations, which other methods can use. An instance of the `DeleteContext` class packages these parameters into an object that can be used when a `deleteRows()` operation is initiated.

IN THIS SECTION:

DeleteContext Properties
DeleteContext Properties

The following are properties for DeleteContext.

IN THIS SECTION:

externalIds
The external IDs of the rows representing external object records to delete.

tableSelected
The name of the table to delete rows from.

**externalIds**
The external IDs of the rows representing external object records to delete.

**Signature**

```java
public List<String> externalIds {get; set;}
```

**Property Value**

Type: List<String>

**tableSelected**
The name of the table to delete rows from.

**Signature**

```java
public String tableSelected {get; set;}
```

**Property Value**

Type: String

DeleteResult Class

Represents the result of a delete operation on an sObject record. The result is returned by the `DataSource.deleteRows` method of the `DataSource.Connection` class.

**Namespace**

`DataSource`

**Usage**

A delete operation on external object records generates an array of objects of type `DataSource.DeleteResult`. Its methods create result records that indicate whether the delete operation succeeded or failed.
DeleteResult Properties

The following are properties for DeleteResult.

errorMessage
The error message that’s generated by a failed delete operation. Recorded with a result of type DataSource.DeleteResult.

externalId
The unique identifier of a row that represents an external object record to delete.

success
Indicates whether a delete operation succeeded or failed.

Signature
public String errorMessage {get; set;}

Property Value
Type: String

externalId
The unique identifier of a row that represents an external object record to delete.

Signature
public String externalId {get; set;}

Property Value
Type: String

success
Indicates whether a delete operation succeeded or failed.

Signature
public Boolean success {get; set;}
DeleteResult Methods

The following are methods for `DeleteResult`.

**IN THIS SECTION:**

- `equals(obj)`
  Maintains the integrity of lists of type `DeleteResult` by determining the equality of external objects in a list. This method is dynamic and is based on the `equals` method in Java.

- `failure(externalId, errorMessage)`
  Creates a delete result indicating the failure of a delete request for a given external ID.

- `hashCode()`
  Maintains the integrity of lists of type `DeleteResult` by determining the uniqueness of the external object records in a list.

- `success(externalId)`
  Creates a delete result indicating the successful completion of a delete request for a given external ID.

### equals(obj)

Maintains the integrity of lists of type `DeleteResult` by determining the equality of external objects in a list. This method is dynamic and is based on the `equals` method in Java.

**Signature**

```java
public Boolean equals(Object obj)
```

**Parameters**

- `obj`
  Type: `Object`
  External object whose key is to be validated.

For information about the `equals` method, see Using Custom Types in Map Keys and Sets.

**Return Value**

Type: `Boolean`

### failure(externalId, errorMessage)

Creates a delete result indicating the failure of a delete request for a given external ID.

**Signature**

```java
public static DataSource.DeleteResult failure(String externalId, String errorMessage)
```
Parameters

- **externalId**
  - Type: `String`
  - The unique identifier of the sObject record to delete.

- **errorMessage**
  - Type: `String`
  - The reason the delete operation failed.

Return Value

- Type: `DataSource.DeleteResult`
  - Status result of the delete operation.

**hashCode()**

Maintains the integrity of lists of type `DeleteResult` by determining the uniqueness of the external object records in a list.

**Signature**

```java
public Integer hashCode()
```

**Return Value**

- Type: `Integer`

**success(externalId)**

Creates a delete result indicating the successful completion of a delete request for a given external ID.

**Signature**

```java
public static DataSource.DeleteResult success(String externalId)
```

**Parameters**

- **externalId**
  - Type: `String`
  - The unique identifier of the sObject record to delete.

**Return Value**

- Type: `DataSource.DeleteResult`
  - Status result of the delete operation for the sObject with the given external ID.

**Filter Class**

Represents a `WHERE` clause in a SOSL or SOQL query.
Namespace
DataSource

Usage
Compound types require child filters. Specifically, the subfilters property can’t be null if the type property is NOT_, AND_, or OR_.

IN THIS SECTION:
Filter Properties

Filter Properties
The following are properties for Filter.

IN THIS SECTION:
columnName
Name of the column that’s being evaluated in a simple comparative type of filter.
columnValue
Value that the filter compares records against in a simple comparative type of filter.
subfilters
List of subfilters for compound filter types, such as NOT_, AND_, and OR_.
tableName
Name of the table whose column is being evaluated in a simple comparative type of filter.
type
Type of filter operation that limits the returned data.

columnName
Name of the column that’s being evaluated in a simple comparative type of filter.

Signature
public String columnName {get; set;}

Property Value
Type: String

columnValue
Value that the filter compares records against in a simple comparative type of filter.
Signature
public Object columnValue {get; set;}

Property Value
Type: Object

subfilters
List of subfilters for compound filter types, such as NOT_, AND_, and OR_.

Signature
public List<DataSource.Filter> subfilters {get; set;}

Property Value
Type: List<DataSource.Filter>

tableName
Name of the table whose column is being evaluated in a simple comparative type of filter.

Signature
public String tableName {get; set;}

Property Value
Type: String

type
Type of filter operation that limits the returned data.

Signature
public DataSource.FilterType type {get; set;}

Property Value
Type: DataSource.FilterType

FilterType Enum
Referenced by the type property on a DataSource.Filter.

Usage
Determines how to limit the returned data.
Enum Values

The following are the values of the `DataSource.FilterType` enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AND</td>
<td>This compound filter type returns all rows that match all the subfilters.</td>
</tr>
<tr>
<td>CONTAINS</td>
<td>Simple comparative filter type.</td>
</tr>
<tr>
<td>ENDS_WITH</td>
<td>Simple comparative filter type.</td>
</tr>
<tr>
<td>EQUALS</td>
<td>Simple comparative filter type.</td>
</tr>
<tr>
<td>GREATER_THAN</td>
<td>Simple comparative filter type.</td>
</tr>
<tr>
<td>GREATER_THAN_OR_EQUAL_TO</td>
<td>Simple comparative filter type.</td>
</tr>
<tr>
<td>LESS_THAN</td>
<td>Simple comparative filter type.</td>
</tr>
<tr>
<td>LESS_THAN_OR_EQUAL_TO</td>
<td>Simple comparative filter type.</td>
</tr>
<tr>
<td>LIKE</td>
<td>Simple comparative filter type.</td>
</tr>
<tr>
<td>NOT</td>
<td>This compound filter type returns the rows that don’t match the subfilter.</td>
</tr>
<tr>
<td>NOT_EQUALS</td>
<td>Simple comparative filter type.</td>
</tr>
<tr>
<td>OR</td>
<td>This compound filter type returns all rows that match any of the subfilters.</td>
</tr>
<tr>
<td>STARTS_WITH</td>
<td>Simple comparative filter type.</td>
</tr>
</tbody>
</table>

IdentityType Enum

Determines which set of credentials is used to authenticate to the external system.

Usage

The relevant credentials are passed to your `DataSource.Connection` class.

Enum Values

The following are the values of the `DataSource.IdentityType` enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANONYMOUS</td>
<td>No credentials are used to authenticate to the external system.</td>
</tr>
<tr>
<td>NAMED_USER</td>
<td>The credentials in the external data source definition are used to authenticate to the external system, regardless of which user is accessing the external data from your organization.</td>
</tr>
</tbody>
</table>
For queries and searches, the credentials are specific to the current user who invokes the query or search. The credentials come from the user’s authentication settings for the external system.

For administrative connections, such as syncing the external system’s schema, the credentials come from the external data source definition.

### Order Class

Contains details about how to sort the rows in the result set. Equivalent to an ORDER BY statement in a SOQL query.

### Namespace

**DataSource**

### Usage

Used in the `order` property on the `DataSource.TableSelection` class.

### Order Properties

The following are properties for `Order`.

#### columnName

Name of the column whose values are used to sort the rows in the result set.

#### direction

Direction for sorting rows based on column values.

#### tableName

Name of the table whose column values are used to sort the rows in the result set.

#### columnName

Name of the column whose values are used to sort the rows in the result set.

### Signature

```java
public String columnName {get; set;}
```
Property Value
type: String

direction
Direction for sorting rows based on column values.

Signature
public DataSource.OrderDirection direction {get; set;}

Property Value
type: DataSource.OrderDirection

tableName
Name of the table whose column values are used to sort the rows in the result set.

Signature
public String tableName {get; set;}

Property Value
type: String

Order Methods
The following are methods for Order.

IN THIS SECTION:
get(tableName, columnName, direction)
Creates an instance of the DataSource.Order class.

get(tableName, columnName, direction)
Creates an instance of the DataSource.Order class.

Signature
public static DataSource.Order get(String tableName, String columnName, DataSource.OrderDirection direction)

Parameters
tableName
type: String
Name of the table whose column values are used to sort the rows in the result set.
**columnName**
Type: String
Name of the column whose values are used to sort the rows in the result set.

**direction**
Type: DataSource.OrderDirection
Direction for sorting rows based on column values.

**Return Value**
Type: DataSource.Order

**OrderDirection Enum**
Specifies the direction for sorting rows based on column values.

**Usage**
Used by the direction property on the DataSource.Order class.

**Enum Values**
The following are the values of the DataSource.OrderDirection enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASCENDING</td>
<td>Sort rows in ascending order (A–Z).</td>
</tr>
<tr>
<td>DESCENDING</td>
<td>Sort rows in descending order (Z–A).</td>
</tr>
</tbody>
</table>

**Provider Class**
Extend this base class to create a custom adapter for Salesforce Connect. The class informs Salesforce of the functional and authentication capabilities that are supported by or required to connect to the external system. This class extends the DataSourceUtil class and inherits its methods.

**Namespace**
DataSource

**Usage**
Create an Apex class that extends DataSource.Provider to specify the following.

- The types of authentication that can be used to access the external system
- The features that are supported for the connection to the external system
- The Apex class that extends DataSource.Connection to sync the external system’s schema and to handle the queries and searches of the external data
The values that are returned by the `DataSource.Provider` class determine which settings are available in the external data source definition in Salesforce. To access the external data source definition from Setup, enter `External Data Sources` in the Quick Find box, then select `External Data Sources`.

**IN THIS SECTION:**

**Provider Methods**

The following are methods for `Provider`.

**IN THIS SECTION:**

- `getAuthenticationCapabilities()`
  Returns the types of authentication that can be used to access the external system.

- `getCapabilities()`
  Returns the functional operations that the external system supports and the required endpoint settings for the external data source definition in Salesforce.

- `getConnection(connectionParams)`
  Returns a connection that points to an instance of the external data source.

---

**getAuthenticationCapabilities()**

Returns the types of authentication that can be used to access the external system.

**Signature**

```java
public List<DataSource.AuthenticationCapability> getAuthenticationCapabilities()
```

**Return Value**

Type: `List<DataSource.AuthenticationCapability>`

---

**getCapabilities()**

Returns the functional operations that the external system supports and the required endpoint settings for the external data source definition in Salesforce.

**Signature**

```java
public List<DataSource.Capability> getCapabilities()
```

**Return Value**

Type: `List<DataSource.Capability>`
getConnection(connectionParams)

Returns a connection that points to an instance of the external data source.

**Signature**

```
public DataSource.Connection getConnection(DataSource.ConnectionParams connectionParams)
```

**Parameters**

- `connectionParams`  
  
  **Type:** DataSource.ConnectionParams  
  
  Credentials for authenticating to the external system.

**Return Value**

- **Type:** DataSource.Connection

**QueryAggregation Enum**

Specifies how to aggregate a column in a query.

**Usage**

Used by the `aggregation` property on the DataSource.ColumnSelection class.

**Enum Values**

The following are the values of the `DataSource.QueryAggregation` enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AVG</td>
<td>Reserved for future use.</td>
</tr>
<tr>
<td>COUNT</td>
<td>Returns the number of rows that meet the query criteria.</td>
</tr>
<tr>
<td>MAX</td>
<td>Reserved for future use.</td>
</tr>
<tr>
<td>MIN</td>
<td>Reserved for future use.</td>
</tr>
<tr>
<td>NONE</td>
<td>No aggregation.</td>
</tr>
<tr>
<td>SUM</td>
<td>Reserved for future use.</td>
</tr>
</tbody>
</table>

**QueryContext Class**

An instance of `QueryContext` is provided to the `query` method on your `DataSource.Connection` class. The instance corresponds to a SOQL request.
Namespace

DataSource

IN THIS SECTION:
QueryContext Properties
QueryContext Methods

QueryContext Properties
The following are properties for QueryContext.

IN THIS SECTION:
queryMoreToken
Query token that’s used for server-driven paging to determine and fetch the subsequent batch of results.
tableSelection
Query details that represent the FROM, ORDER BY, SELECT, and WHERE clauses in a SOQL or SOSL query.

queryMoreToken
Query token that’s used for server-driven paging to determine and fetch the subsequent batch of results.

Signature
public String queryMoreToken {get; set;}

Property Value
Type: String

tableSelection
Query details that represent the FROM, ORDER BY, SELECT, and WHERE clauses in a SOQL or SOSL query.

Signature
public DataSource.TableSelection tableSelection {get; set;}

Property Value
Type: DataSource.TableSelection

QueryContext Methods
The following are methods for QueryContext.
IN THIS SECTION:

get(metadata, offset, maxResults, tableSelection)
Creates an instance of the QueryContext class.

get(metadata, offset, maxResults, tableSelection)
Creates an instance of the QueryContext class.

Signature

public static DataSource.QueryContext get(List<DataSource.Table> metadata, Integer offset, Integer maxResults, DataSource.TableSelection tableSelection)

Parameters

metadata
Type: List<DataSource.Table>
List of table metadata that describes the external system’s tables to query.

dataoffset
Type: Integer
Used for client-driven paging. Specifies the starting row offset into the query’s result set.

maxResults
Type: Integer
Used for client-driven paging. Specifies the maximum number of rows to return in each batch.

tableSelection
Type: DataSource.TableSelection
Query details that represent the FROM, ORDER BY, SELECT, and WHERE clauses in a SOQL or SOSL query.

Return Value

Type: DataSource.QueryContext

QueryUtils Class

Contains helper methods to locally filter, sort, and apply limit and offset clauses to data rows. This helper class is provided for your convenience during early development and tests, but it isn’t supported for use in production environments.

Namespace

DataSource

Usage

The DataSource.QueryUtils class and its helper methods can process query results locally within your Salesforce org. This class is provided for your convenience to simplify the development of your Salesforce Connect custom adapter for initial tests. However, the DataSource.QueryUtils class and its methods aren’t supported for use in production environments that use callouts to retrieve...
data from external systems. Complete the filtering and sorting on the external system before sending the query results to Salesforce. When possible, use server-driven paging or another technique to have the external system determine the appropriate data subsets according to the limit and offset clauses in the query.

IN THIS SECTION:

QueryUtils Methods

QueryUtils Methods

The following are methods for QueryUtils.

IN THIS SECTION:

applyLimitAndOffset(queryContext, rows)

Returns a subset of data rows after locally applying limit and offset clauses from the query. This helper method is provided for your convenience during early development and tests, but it isn’t supported for use in production environments.

filter(queryContext, rows)

Returns a subset of data rows after locally ordering and applying filters from the query. This helper method is provided for your convenience during early development and tests, but it isn’t supported for use in production environments.

process(queryContext, rows)

Returns data rows after locally filtering, sorting, ordering, and applying limit and offset clauses from the query. This helper method is provided for your convenience during early development and tests, but it isn’t supported for use in production environments.

sort(queryContext, rows)

Returns data rows after locally sorting and applying the order from the query. This helper method is provided for your convenience during early development and tests, but it isn’t supported for use in production environments.

applyLimitAndOffset(queryContext, rows)

Returns a subset of data rows after locally applying limit and offset clauses from the query. This helper method is provided for your convenience during early development and tests, but it isn’t supported for use in production environments.

Signature

public static List<Map<String, Object>> applyLimitAndOffset(DataSource.QueryContext queryContext, List<Map<String, Object>> rows)

Parameters

queryContext

  Type: DataSource.QueryContext

  Represents the query to run against a data table.

rows

  Type: List<Map<String, Object>>

  Rows of data.
Return Value
Type: List<Map<String, Object>>

**filter** (queryContext, rows)
Returns a subset of data rows after locally ordering and applying filters from the query. This helper method is provided for your convenience during early development and tests, but it isn’t supported for use in production environments.

**Signature**

```java
public static List<Map<String, Object>> filter(DataSource.QueryContext queryContext, List<Map<String, Object>> rows)
```

**Parameters**

- **queryContext**
  Type: DataSource.QueryContext
- **rows**
  Type: List<Map<String, Object>>

  Rows of data.

Return Value
Type: List<Map<String, Object>>

**process** (queryContext, rows)
Returns data rows after locally filtering, sorting, ordering, and applying limit and offset clauses from the query. This helper method is provided for your convenience during early development and tests, but it isn’t supported for use in production environments.

**Signature**

```java
public static List<Map<String, Object>> process(DataSource.QueryContext queryContext, List<Map<String, Object>> rows)
```

**Parameters**

- **queryContext**
  Type: DataSource.QueryContext
  Represents the query to run against a data table.
- **rows**
  Type: List<Map<String, Object>>

  Rows of data.

Return Value
Type: List<Map<String, Object>>
sort(queryContext, rows)
Returns data rows after locally sorting and applying the order from the query. This helper method is provided for your convenience during early development and tests, but it isn’t supported for use in production environments.

Signature
public static List<Map<String, Object>> sort(DataSource.QueryContext queryContext, List<Map<String, Object>> rows)

Parameters
queryContext
Type: DataSource.QueryContext
Represents the query to run against a data table.

rows
Type: List<Map<String, Object>>
Rows of data.

Return Value
Type: List<Map<String, Object>>

ReadContext Class
Abstract base class for the QueryContext and SearchContext classes.

Namespace
DataSource

IN THIS SECTION:
ReadContext Properties

ReadContext Properties
The following are properties for ReadContext.

IN THIS SECTION:
maxResults
Maximum number of rows that the query can return.
metadata
Describes the external system’s tables to query.
offset
The starting row offset into the query’s result set. Used for client-driven paging.
**maxResults**
Maximum number of rows that the query can return.

**Signature**
```java
public Integer maxResults {get; set;}
```

**Property Value**
Type: Integer

**metadata**
Describes the external system’s tables to query.

**Signature**
```java
public List<DataSource.Table> metadata {get; set;}
```

**Property Value**
Type: List<DataSource.Table>

**offset**
The starting row offset into the query’s result set. Used for client-driven paging.

**Signature**
```java
public Integer offset {get; set;}
```

**Property Value**
Type: Integer

**SearchContext Class**

An instance of SearchContext is provided to the search method on your DataSource.Connection class. The instance corresponds to a search or SOSL request.

**Namespace**
```
DataSource
```

IN THIS SECTION:
- SearchContext Constructors
- SearchContext Properties
SearchContext Constructors

The following are constructors for SearchContext.

IN THIS SECTION:

SearchContext(metadata, offset, maxResults, tableSelections, searchPhrase)
Creates an instance of the SearchContext class with the specified parameter values.

SearchContext()
Creates an instance of the SearchContext class.

SearchContext (metadata, offset, maxResults, tableSelections, searchPhrase)
Creates an instance of the SearchContext class with the specified parameter values.

Signature

public SearchContext(List<DataSource.Table> metadata, Integer offset, Integer maxResults, List<DataSource.TableSelection> tableSelections, String searchPhrase)

Parameters

metadata
Type: List<DataSource.Table>
List of table metadata that describes the external system’s tables to query.

offset
Type: Integer
Specifies the starting row offset into the query’s result set.

maxResults
Type: Integer
Specifies the maximum number of rows to return in each batch.

tableSelections
Type: List<DataSource.TableSelection>
List of queries and their details. The details represent the FROM, ORDER BY, SELECT, and WHERE clauses in each SOQL or SOSL query.

searchPhrase
Type: String
The user-entered search string as a case-sensitive single phrase, with all non-alphanumeric characters removed.

SearchContext ()
Creates an instance of the SearchContext class.

Signature

public SearchContext ()
SearchContext Properties

The following are properties for SearchContext.

**IN THIS SECTION:**

**searchPhrase**

The user-entered search string as a case-sensitive single phrase, with all non-alphanumeric characters removed.

**tableSelections**

List of queries and their details. The details represent the FROM, ORDER BY, SELECT, and WHERE clauses in each SOQL or SOSL query.

**SearchPhrase**

The user-entered search string as a case-sensitive single phrase, with all non-alphanumeric characters removed.

**Signature**

```java
public String searchPhrase {get; set;}
```

**Property Value**

Type: String

**tableSelections**

List of queries and their details. The details represent the FROM, ORDER BY, SELECT, and WHERE clauses in each SOQL or SOSL query.

**Signature**

```java
public List<DataSource.TableSelection> tableSelections {get; set;}
```

**Property Value**

Type: List<DataSource.TableSelection>

**SearchUtils Class**

Helper class for implementing search on a custom adapter for Salesforce Connect.

**Namespace**

DataSource

**Usage**

We recommend that you develop your own search implementation that can search columns in addition to the designated name field.

**IN THIS SECTION:**

**SearchUtils Methods**
SearchUtils Methods

The following are methods for SearchUtils.

IN THIS SECTION:

searchByName(searchDetails, connection)

Queries all the tables and returns each row whose designated name field contains the search phrase.

searchByName(searchDetails, connection)

Queries all the tables and returns each row whose designated name field contains the search phrase.

**Signature**

`public static List<DataSource.TableResult> searchByName(DataSource.SearchContext searchDetails, DataSource.Connection connection)`

**Parameters**

- `searchDetails`
  Type: `DataSource.SearchContext`
  The `SearchContext` class that specifies which data to search and what to search for.

- `connection`
  Type: `DataSource.Connection`
  The `DataSource.Connection` class that connects to the external system.

**Return Value**

Type: `List<DataSource.TableResult>`

**Table Class**

Describes a table on an external system that the Salesforce Connect custom adapter connects to. This class extends the `DataSourceUtil` class and inherits its methods.

**Namespace**

`DataSource`

**Usage**

A list of table metadata is provided by the `DataSource.Connection` class when the `sync()` method is invoked. Each table can become an external object in Salesforce.

The metadata is stored in Salesforce. Updating the Apex code to return new or updated values for the table metadata doesn’t automatically update the stored metadata in Salesforce.
Table Properties

The following are properties for Table.

IN THIS SECTION:

columns
List of table columns.

description
Description of what the table represents.

description
Plural form of the user-friendly name for the table. The labelPlural becomes the object's plural label in the Salesforce user interface.

description
Singular form of the user-friendly name for the table. The labelSingular becomes the object label in the Salesforce user interface. We recommend that you make object labels unique across all standard, custom, and external objects in the org.

name
Name of the table on the external system.

nameColumn
Name of the table column that becomes the name field of the external object when the administrator syncs the table.

columns
List of table columns.

Signature

public List<DataSource.Column> columns {get; set;}

Property Value

Type: List<DataSource.Column>

description
Description of what the table represents.

Signature

public String description {get; set;
Property Value
Type: String

**labelPlural**
Plural form of the user-friendly name for the table. The `labelPlural` becomes the object's plural label in the Salesforce user interface.

**Signature**

```java
public String labelPlural {get; set;}
```

Property Value
Type: String

**labelSingular**
Singular form of the user-friendly name for the table. The `labelSingular` becomes the object label in the Salesforce user interface. We recommend that you make object labels unique across all standard, custom, and external objects in the org.

**Signature**

```java
public String labelSingular {get; set;}
```

Property Value
Type: String

**name**
Name of the table on the external system.

**Signature**

```java
public String name {get; set;}
```

Property Value
Type: String

**nameColumn**
Name of the table column that becomes the name field of the external object when the administrator syncs the table.

**Signature**

```java
public String nameColumn {get; set;}
```
Property Value
Type: String

Table Methods
The following are methods for Table.

IN THIS SECTION:

get(name, labelSingular, labelPlural, description, nameColumn, columns)
Returns the table metadata with the specified parameter values.

get(name, nameColumn, columns)
Returns the table metadata with the specified parameter values, using the name for the labels and description.

get(name, labelSingular, labelPlural, description, nameColumn, columns)
Returns the table metadata with the specified parameter values.

Signature
public static DataSource.Table get(String name, String labelSingular, String labelPlural, String description, String nameColumn, List<DataSource.Column> columns)

Parameters

name
Type: String
Name of the external table.

labelSingular
Type: String
Singular form of the user-friendly name for the table. The labelSingular becomes the object label in the Salesforce user interface.

labelPlural
Type: String
Plural form of the user-friendly name for the table. The labelPlural becomes the object’s plural label in the Salesforce user interface.

description
Type: String
Description of the external table.

nameColumn
Type: String
Name of the table column that becomes the name field of the external object when the administrator syncs the table.

columns
Type: List<DataSource.Column>
List of table columns.
Return Value
Type: DataSource.Table

get(name, nameColumn, columns)
Returns the table metadata with the specified parameter values, using the name for the labels and description.

Signature
public static DataSource.Table get(String name, String nameColumn, List<DataSource.Column> columns)

Parameters
name
Type: String
Name of the external table.

nameColumn
Type: String
Name of the table column that becomes the name field of the external object when the administrator syncs the table.

columns
Type: List<DataSource.Column>
List of table columns.

Return Value
Type: DataSource.Table
The returned table metadata has these property values.

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>name</td>
</tr>
<tr>
<td>labelSingular</td>
<td>name</td>
</tr>
<tr>
<td>labelPlural</td>
<td>name</td>
</tr>
<tr>
<td>description</td>
<td>name</td>
</tr>
<tr>
<td>nameColumn</td>
<td>nameColumn</td>
</tr>
<tr>
<td>columns</td>
<td>columns</td>
</tr>
</tbody>
</table>

TableResult Class
Contains the results of a search or query.
Namespace

**DataSource**

IN THIS SECTION:

- **TableResult Properties**
- **TableResult Methods**

**TableResult Properties**

The following are properties for TableResult.

IN THIS SECTION:

- **errorMessage**
- **errorMessage**
- **queryMoreToken**
  
  Query token that's used for server-driven paging to determine and fetch the subsequent batch of results. This token is passed back to the Apex data source on subsequent queries in the `queryMoreToken` property on the `QueryContext`.

- **rows**
  
  Rows of data.

- **success**
  
  Whether the search or query was successful.

- **tableName**
  
  Name of the table that was queried.

- **totalSize**
  
  The total number of rows that meet the query criteria, even when the external system is requested to return a smaller batch size.

**errorMessage**

errorMessage

**Signature**

```java
public String errorMessage {get; set;}
```

**Property Value**

Type: **String**

**queryMoreToken**

Query token that's used for server-driven paging to determine and fetch the subsequent batch of results. This token is passed back to the Apex data source on subsequent queries in the `queryMoreToken` property on the `QueryContext`. 
Signature
public String queryMoreToken {get; set;}

Property Value
Type: String

rows
Rows of data.

Signature
public List<Map<String, Object>> rows {get; set;}

Property Value
Type: List<Map<String, Object>>

success
Whether the search or query was successful.

Signature
public Boolean success {get; set;}

Property Value
Type: Boolean

tableName
Name of the table that was queried.

Signature
public String tableName {get; set;}

Property Value
Type: String

totalSize
The total number of rows that meet the query criteria, even when the external system is requested to return a smaller batch size.

Signature
public Integer totalSize {get; set;}
**Property Value**

Type: Integer

**TableResult Methods**

The following are methods for `TableResult`.

**IN THIS SECTION:**

- `error(errorMessage)`
  Returns failed search or query results with the provided error message.
- `get(success, errorMessage, tableName, rows, totalSize)`
  Returns a subset of data rows in a `TableResult` with the provided property values and the number of rows in the table.
- `get(success, errorMessage, tableName, rows)`
  Returns a subset of data rows in a `TableResult` with the provided property values.
- `get(queryContext, rows)`
  Returns the subset of data rows that meet the query criteria, and the number of rows in the table, in a `TableResult`.
- `get(tableSelection, rows)`
  Returns the subset of data rows that meet the query criteria, and the number of rows in the table, in a `TableResult`.

**error(errorMessage)**

Returns failed search or query results with the provided error message.

**Signature**

```java
public static DataSource.TableResult error(String errorMessage)
```

**Parameters**

- `errorMessage`
  Type: String
  errorMessage

**Return Value**

Type: `DataSource.TableResult`

The returned `TableResult` has these property values.

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>success</td>
<td>false</td>
</tr>
<tr>
<td>errorMessage</td>
<td><code>errorMessage</code></td>
</tr>
<tr>
<td>tableName</td>
<td>null</td>
</tr>
<tr>
<td>rows</td>
<td>null</td>
</tr>
</tbody>
</table>
get(success, errorMessage, tableName, rows, totalSize)

Returns a subset of data rows in a TableResult with the provided property values and the number of rows in the table.

Signature

```java
public static DataSource.TableResult get(Boolean success, String errorMessage, String tableName, List<Map<String, Object>> rows, Integer totalSize)
```

Parameters

- **success**
  - Type: Boolean
  - Whether the search or query was successful.
- **errorMessage**
  - Type: String
  - errorMessage
- **tableName**
  - Type: String
  - Name of the table that was queried.
- **rows**
  - Type: List<Map<String, Object>>
  - Rows of data.
- **totalSize**
  - Type: Integer
  - The total number of rows that meet the query criteria, even when the external system is requested to return a smaller batch size.

Return Value

Type: DataSource.TableResult

get(success, errorMessage, tableName, rows)

Returns a subset of data rows in a TableResult with the provided property values.

Signature

```java
public static DataSource.TableResult get(Boolean success, String errorMessage, String tableName, List<Map<String, Object>> rows)
```
Parameters

success
  Type: Boolean
  Whether the search or query was successful.

errorMessage
  Type: String

tableName
  Type: String
  Name of the table that was queried.

rows
  Type: List<Map<String, Object>>
  Rows of data.

Return Value
Type: DataSource.TableResult

get(queryContext, rows)
Returns the subset of data rows that meet the query criteria, and the number of rows in the table, in a TableResult.

Signature
public static DataSource.TableResult get(DataSource.QueryContext queryContext, List<Map<String, Object>> rows)

Parameters

queryContext
  Type: DataSource.QueryContext
  Represents the query to run against a data table.

rows
  Type: List<Map<String, Object>>
  Rows of data.

Return Value
Type: DataSource.TableResult

get(tableName, rows)
Returns the subset of data rows that meet the query criteria, and the number of rows in the table, in a TableResult.
Signature

public static DataSource.TableResult get(DataSource.TableSelection tableSelection, List<Map<String, Object>> rows)

Parameters

   tableSelection
     Type: DataSource.TableSelection
     Query details that represent the FROM, ORDER BY, SELECT, and WHERE clauses in a SOQL or SOSL query.

   rows
     Type: List<Map<String, Object>>
     Rows of data.

Return Value

Type: DataSource.TableResult

TableSelection Class

Contains a breakdown of the SOQL or SOSL query. Its properties represent the FROM, ORDER BY, SELECT, and WHERE clauses in the query.

Namespace

DataSource

IN THIS SECTION:
   TableSelection Properties

TableSelection Properties

The following are properties for TableSelection.

IN THIS SECTION:
   columnsSelected
     List of columns to query. Corresponds to the SELECT clause in a SOQL or SOSL query.

   filter
     Identifies the query filter, which can be a compound filter that has a list of subfilters. The filter corresponds to the WHERE clause in a SOQL or SOSL query.

   order
     Identifies the order for sorting the query results. Corresponds to the ORDER BY clause in a SOQL or SOSL query.

   tableSelected
     Name of the table to query. Corresponds to the FROM clause in a SOQL or SOSL query.
**columnsSelected**
List of columns to query. Corresponds to the **SELECT** clause in a SOQL or SOSL query.

**Signature**
```csharp
public List<DataSource.ColumnSelection> columnsSelected {get; set;}
```

**Property Value**
Type: List<DataSource.ColumnSelection>

**filter**
Identifies the query filter, which can be a compound filter that has a list of subfilters. The filter corresponds to the **WHERE** clause in a SOQL or SOSL query.

**Signature**
```csharp
public DataSource.Filter filter {get; set;}
```

**Property Value**
Type: DataSource.Filter

**order**
Identifies the order for sorting the query results. Corresponds to the **ORDER BY** clause in a SOQL or SOSL query.

**Signature**
```csharp
public List<DataSource.Order> order {get; set;}
```

**Property Value**
Type: List<DataSource.Order>

**tableSelected**
Name of the table to query. Corresponds to the **FROM** clause in a SOQL or SOSL query.

**Signature**
```csharp
public String tableSelected {get; set;}
```

**Property Value**
Type: String
UpsertContext Class

An instance of ` UpsertContext` is passed to the `upsertRows()` method on your `Datasource.Connection` class. This class provides context information about the upsert request to the implementor of `upsertRows()`.

Namespace

`DataSource`

Usage

The Apex Connector Framework creates the context for operations. Context is comprised of parameters about the operations, which other methods can use. An instance of the `UpsertContext` class packages these parameters into an object that can be used when an `upsertRows()` operation is initiated.

IN THIS SECTION:

- `UpsertContext Properties`

UpsertContext Properties

The following are properties for `UpsertContext`.

IN THIS SECTION:

- `rows`
- `tableSelected`

(rows)

List of rows corresponding to the external object records to upsert.

(tableSelected)

The name of the table to upsert rows in.

(rows)

List of rows corresponding to the external object records to upsert.

Signature

```java
public List<Map<String,ANY>> rows {get; set;}
```

Property Value

Type: `List<Map<String,object>>`

(tableSelected)

The name of the table to upsert rows in.

Signature

```java
public String tableSelected {get; set;}
```
UpsertResult Class

Represents the result of an upsert operation on an external object record. The result is returned by the upsertRows method of the DataSource.Connection class.

Namespace

DataSource

Usage

An upsert operation on external object records generates an array of objects of type DataSource.UpsertResult. Its methods create result records that indicate whether the upsert operation succeeded or failed.

IN THIS SECTION:

UpsertResult Properties

UpsertResult Methods

UpsertResult Properties

The following are properties for UpsertResult.

IN THIS SECTION:

errorMessage

The error message that's generated by a failed upsert operation.

externalId

The unique identifier of a row that represents an external object record to upsert.

success

Indicates whether a delete operation succeeded or failed.

errorMessage

The error message that's generated by a failed upsert operation.

Signature

public String errorMessage {get; set;}

Property Value

Type: String
**externalId**
The unique identifier of a row that represents an external object record to upsert.

**Signature**
```java
public String externalId {get; set;}
```

**Property Value**
Type: String

**success**
Indicates whether a delete operation succeeded or failed.

**Signature**
```java
public Boolean success {get; set;}
```

**Property Value**
Type: Boolean

**UpsertResult Methods**
The following are methods for UpsertResult.

**IN THIS SECTION:**
- `equals(obj)`
  Maintains the integrity of lists of type UpsertResult by determining the equality of external object records in a list. This method is dynamic and is based on the `equals` method in Java.
- `failure(externalId, errorMessage)`
  Creates an upsert result that indicates the failure of a delete request for a given external ID.
- `hashCode()`
  Maintains the integrity of lists of type UpsertResult by determining the uniqueness of the external object records in a list.
- `success(externalId)`
  Creates a delete result that indicates the successful completion of an upsert request for a given external ID.

**equals(obj)**
Maintains the integrity of lists of type UpsertResult by determining the equality of external object records in a list. This method is dynamic and is based on the `equals` method in Java.

**Signature**
```java
public Boolean equals(Object obj)
```
Parameters

**obj**
Type: Object
External object whose key is to be validated.

Return Value
Type: Boolean

**failure(externalId, errorMessage)**
Creates an upsert result that indicates the failure of a delete request for a given external ID.

Signature

```
public static DataSource.UpsertResult failure(String externalId, String errorMessage)
```

Parameters

- **externalId**
  Type: String
  The unique identifier of the external object record to upsert.

- **errorMessage**
  Type: String
  The reason the upsert operation failed.

Return Value
Type: DataSource.UpsertResult
Status result for the upsert operation.

**hashCode()**

Maintains the integrity of lists of type UpsertResult by determining the uniqueness of the external object records in a list.

Signature

```
public Integer hashCode()
```

Return Value
Type: Integer

**success(externalId)**
Creates a delete result that indicates the successful completion of an upsert request for a given external ID.
**Signature**

```java
public static DataSource.UpsertResult success(String externalId)
```

**Parameters**

- `externalId`
  - Type: `String`
  - The unique identifier of the external object record to upsert.

**Return Value**

- Type: `DataSource.UpsertResult`
  - Status result of the upsert operation for the external object record with the given external ID.

**DataSource Exceptions**

The `DataSource` namespace contains exception classes. All exception classes support built-in methods for returning the error message and exception type. See Exception Class and Built-In Exceptions.

The `DataSource` namespace contains these exceptions.

<table>
<thead>
<tr>
<th>Exception</th>
<th>Description</th>
<th>Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>DataSource.DataSourceException</code></td>
<td>Throw this exception to indicate that an error occurred while communicating with an external data source.</td>
<td>To get the error message and write it to debug log, use the <code>String getMessage()</code> method.</td>
</tr>
<tr>
<td><code>DataSource.OAuthTokenExpiredException</code></td>
<td>Throw this exception to indicate that an OAuth token has expired. The system then attempts to refresh the token automatically and restart the query, search, or sync operation.</td>
<td>To get the error message and write it to debug log, use the <code>String getMessage()</code> method.</td>
</tr>
</tbody>
</table>

**DataWeave Namespace**

The `DataWeave` namespace provides classes and methods to support the invocation of `DataWeave` scripts from Apex.

`DataWeave` is the MuleSoft expression language for accessing, parsing, and transforming data that travels through a Mule application.

For detailed information, see DataWeave Language.

These are the classes in the `DataWeave` namespace.

**IN THIS SECTION:**

- **Result Class**
  - Contains methods to retrieve data that was transformed using `Script` class methods.
Script Class
Contains the `createScript()` method to load DataWeave scripts and the `execute()` method to obtain script output in a `DataWeave.Result` object.

SEE ALSO:
DataWeave in Apex

Result Class
Contains methods to retrieve data that was transformed using Script class methods.

Namespace
DataWeave

Example
See Script Class for an example to run a DataWeave script from Apex and retrieve the resulting script output.

IN THIS SECTION:
Result Methods

Result Methods
The following are methods for Result.

IN THIS SECTION:
getValue()
Returns the result of a DataWeave script execution as an object.

getValueAsString()
Returns the result of a DataWeave script execution as a string value.

```
getValue()
```
Returns the result of a DataWeave script execution as an object.

**Signature**

```
public Object getValue()
```

**Return Value**
Type: Object
getValueAsString()

Returns the result of a DataWeave script execution as a string value.

**Signature**

```java
public String getValueAsString()
```

**Return Value**

Type: String

**Script Class**

Contains the `createScript()` method to load DataWeave scripts and the `execute()` method to obtain script output in a `DataWeave.Result` object.

**Namespace**

DataWeave

This example runs a DataWeave script from Apex and retrieves the resulting script output. First deploy the script to the org as `ContactsToJson.dwl`.

```java
%dw 2.0
input records application/java
output application/json
---
{
  users: records map(record) -> {
    firstName: record.FirstName,
    lastName: record.LastName
  }
}
```

Then, execute the script from Apex.

```java
List<Contact> data = [SELECT FirstName, LastName FROM Contact WHERE LastName LIMIT 5];
Map<String, Object> args = new Map<String, Object>{ 'records' => data };
DataWeave.Script script = DataWeave.Script.createScript('ContactsToJson');

DataWeave.Result result = script.execute(args);
string jsonOutput = result.getValueAsString();
```

**IN THIS SECTION:**

Script Methods

The following are methods for `Script`. 

**Script Methods**

The following are methods for `Script`. 

---

2181
IN THIS SECTION:

createScript(scriptName)
Loads a DataWeave 2.0 script from the .dwl metadata file that is deployed in an org. The script can then be run using the Script.execute method.

createScript(namespace, scriptName)
Loads a DataWeave 2.0 script from a specified namespace. The script can then be run using the Script.execute method.

execute(parameters)
Executes the DataWeave script that is loaded using the createScript() method and returns the script output.

toString()
Returns the name of the script.

createScript(scriptName)
Loads a DataWeave 2.0 script from the .dwl metadata file that is deployed in an org. The script can then be run using the Script.execute method.

Signature

class public static createScript(String scriptName)

Parameters

scriptName
Type: String
The name of the deployed metadata .dwl script (not including the file extension).

Return Value

Type: DataWeave.Script
DataWeave script that is used as a parameter in the Script.execute() method.

createScript(namespace, scriptName)
Loads a DataWeave 2.0 script from a specified namespace. The script can then be run using the Script.execute method.

Signature

class public static dataweave.Script createScript(String namespace, String scriptName)

Parameters

namespace
Type: String
The namespace name for the deployed script. If the namespace name is null, the caller namespace is used. If the namespace name is empty, the org namespace is used.

scriptName
Type: String
The name of the deployed metadata .dw1 script (not including the file extension).

**Return Value**
Type: DataWeave.Script
DataWeave script that is used as a parameter in the Script.execute() method.

**execute (parameters)**
Executes the DataWeave script that is loaded using the createScript() method and returns the script output.

**Signature**
```java
public execute(Map<String, Object> parameters)
```

**Parameters**
- `parameters`
  - Type: Map<String, Object>
  - Input to the DataWeave script. The keys correspond to the input directive names defined in the DataWeave header. See Input Directive and DataWeave Header.

**Return Value**
Type: DataWeave.Result
The DataWeave.Result object contains the script output.

**toString()**
Returns the name of the script.

**Signature**
```java
public String toString()
```

**Return Value**
Type: String

---

**Dom Namespace**
The Dom namespace provides classes and methods for parsing and creating XML content. The following are the classes in the Dom namespace.

**IN THIS SECTION:**
- Document Class
  - Use the Document class to process XML content. You can parse nested XML content that’s up to 50 nodes deep.
XmlNode Class
Use the XmlNode class to work with a node in an XML document.

Document Class
Use the Document class to process XML content. You can parse nested XML content that's up to 50 nodes deep.

Namespace
Dom

Usage
One common application is to use it to create the body of a request for HttpRequest or to parse a response accessed by HttpResponse.

IN THIS SECTION:
  Document Constructors
  Document Methods

SEE ALSO:
  Reading and Writing XML Using the DOM

Document Constructors
The following are constructors for Document.

IN THIS SECTION:
  Document()
  Document()


Signature

public Document()
getRootElement()
Returns the top-level root element node in the document. If this method returns \texttt{null}, the root element has not been created yet.

load(xml)
Parse the XML representation of the document specified in the \texttt{xml} argument and load it into a document.

toXmlString()
Returns the XML representation of the document as a String.

createRootElement(name, namespace, prefix)
Creates the top-level root element for a document.

\textbf{Signature}

\texttt{public Dom.XmlNode createRootElement(String name, String namespace, String prefix)}

\textbf{Parameters}

\textit{name}
Type: String

\textit{namespace}
Type: String

\textit{prefix}
Type: String

\textbf{Return Value}
Type: Dom.XmlNode

\textbf{Usage}
For more information about namespaces, see \textit{Reading and Writing XML Using the DOM}.
Calling this method more than once on a document generates an error as a document can have only one root element.

gRootElement()
Returns the top-level root element node in the document. If this method returns \texttt{null}, the root element has not been created yet.

\textbf{Signature}

\texttt{public Dom.XmlNode getRootElement()}

\textbf{Return Value}
Type: Dom.XmlNode

load(xml)
Parse the XML representation of the document specified in the \texttt{xml} argument and load it into a document.
Signature
public Void load(String xml)

Parameters
xml
Type: String

Return Value
Type: Void

Example
doc.load(xml);

toXmlString()
Returns the XML representation of the document as a String.

Signature
public String toXmlString()

Return Value
Type: String

XmlNode Class
Use the XmlNode class to work with a node in an XML document.

Namespace
Dom

XmlNode Methods
The following are methods for XmlNode. All are instance methods.

IN THIS SECTION:
   addChildElement(name, namespace, prefix)
   Creates a child element node for this node.
   addCommentNode(text)
   Creates a child comment node for this node.
addTextNode(text)
 creates a child text node for this node.

getAttribute(key, keyNamespace)
 returns namespace:prefix:attributeValue for the given key and key namespace.

getAttributeCount()
 returns the number of attributes for this node.

getAttributeKeyAt(index)
 returns the attribute key for the given index. Index values start at 0.

getAttributeKeyNsAt(index)
 returns the attribute key namespace for the given index.

getAttributeValue(key, keyNamespace)
 returns the attribute value for the given key and key namespace.

getAttributeValueNs(key, keyNamespace)
 returns the attribute value namespace for the given key and key namespace.

getChildElement(name, namespace)
 returns the child element node for the node with the given name and namespace.

getChildElements()
 returns the child element nodes for this node. This doesn’t include child text or comment nodes.

getChildren()
 returns the child nodes for this node. This includes all node types.

getName()
 returns the element name.

getNamespace()
 returns the namespace of the element.

getNamespaceFor(prefix)
 returns the namespace of the element for the given prefix.

getNodeType()
 returns the node type.

getParent()
 returns the parent of this element.

getPrefixFor(namespace)
 returns the prefix of the given namespace.

getText()
 returns the text for this node.

insertBefore(newChild, refChild)
 inserts a new child node before the specified node.

removeAttribute(key, keyNamespace)
 removes the attribute with the given key and key namespace. Returns true if successful, false otherwise.

removeChild(childNode)
 removes the given child node.
setAttribute(key, value)
Sets the key attribute value.
setAttributeNs(key, value, keyNamespace, valueNamespace)
Sets the key attribute value.
setNamespace(prefix, namespace)
Sets the namespace for the given prefix.

addChildElement(name, namespace, prefix)
Creates a child element node for this node.

Signature
public Dom.XmlNode addChildElement(String name, String namespace, String prefix)

Parameters
name
Type: String
The name argument can’t have a null value.
namespace
Type: String
prefix
Type: String

Return Value
Type: Dom.XmlNode

Usage
• If the namespace argument has a non-null value and the prefix argument is null, the namespace is set as the default namespace.
• If the prefix argument is null, Salesforce automatically assigns a prefix for the element. The format of the automatic prefix is ns$i, where i is a number. If the prefix argument is '', the namespace is set as the default namespace.

addCommentNode(text)
Creates a child comment node for this node.

Signature
public Dom.XmlNode addCommentNode(String text)

Parameters
text
Type: String
The `text` argument can't have a `null` value.

Return Value
Type: `Dom.XmlNode`

`addTextNode(text)`
Creates a child text node for this node.

Signature
```
public Dom.XmlNode addTextNode(String text)
```

Parameters
`text`
Type: `String`
The `text` argument can't have a `null` value.

Return Value
Type: `Dom.XmlNode`

`getAttribute(key, keyNamespace)`
Returns `namespacePrefix:attributeValue` for the given key and key namespace.

Signature
```
public String getAttribute(String key, String keyNamespace)
```

Parameters
`key`
Type: `String`
`keyNamespace`
Type: `String`

Return Value
Type: `String`

Example
For example, for the `<xyz a:b="c:d" />` element:

- `getAttribute` returns `c:d`
- `getAttributeValue` returns `d`
getAttributeCount()
Returns the number of attributes for this node.

Signature
public Integer getAttributeCount()

Return Value
Type: Integer

ggetAttributeKeyAt(index)
Returns the attribute key for the given index. Index values start at 0.

Signature
public String getAttributeKeyAt(Integer index)

Parameters
index
Type: Integer

Return Value
Type: String

ggetAttributeKeyNsAt(index)
Returns the attribute key namespace for the given index.

Signature
public String getAttributeKeyNsAt(Integer index)

Parameters
index
Type: Integer

Return Value
Type: String

ggetAttributeValue(key, keyNamespace)
Returns the attribute value for the given key and key namespace.
**getAttributeValue(key, keyNamespace)**

```java
public String getAttributeValue(String key, String keyNamespace)
```

**Parameters**

- `key`
  - Type: `String`
- `keyNamespace`
  - Type: `String`

**Return Value**

Type: `String`

**Example**

For example, for the `<xyz a:b="c:d" />` element:

- `getAttribute` returns `c:d`
- `getAttributeValue` returns `d`

**getAttributeValueNs(key, keyNamespace)**

Returns the attribute value namespace for the given key and key namespace.

**Signature**

```java
public String getAttributeValueNs(String key, String keyNamespace)
```

**Parameters**

- `key`
  - Type: `String`
- `keyNamespace`
  - Type: `String`

**Return Value**

Type: `String`

**getChildElement(name, namespace)**

Returns the child element node for the node with the given name and namespace.

**Signature**

```java
public Dom.XmlNode getChildElement(String name, String namespace)
```
Parameters

name
Type: String
namespace
Type: String

Return Value
Type: Dom.XmlNode

**getChildElements()**
Returns the child element nodes for this node. This doesn't include child text or comment nodes.

Signature

```java
public Dom.XmlNode[] getChildElements()
```

Return Value
Type: Dom.XmlNode[]

**getChildren()**
Returns the child nodes for this node. This includes all node types.

Signature

```java
public Dom.XmlNode[] getChildren()
```

Return Value
Type: Dom.XmlNode[]

**getName()**
Returns the element name.

Signature

```java
public String getName()
```

Return Value
Type: String

**getNamespace()**
Returns the namespace of the element.
Signature

public String getNamespace()

Return Value
Type: String

getNamespaceFor(prefix)
Returns the namespace of the element for the given prefix.

Signature

public String getNamespaceFor(String prefix)

Parameters
prefix
Type: String

Return Value
Type: String

getNodeType()
Returns the node type.

Signature

public Dom.XmlNodeType getNodeType()

Return Value
Type: Dom.XmlNodeType

getParent()
Returns the parent of this element.

Signature

public Dom.XmlNode getParent()

Return Value
Type: Dom.XmlNode
**getPrefixFor(namespace)**
Returns the prefix of the given namespace.

**Signature**
```java
public String getPrefixFor(String namespace)
```

**Parameters**
- `namespace`
  - Type: `String`
  - The `namespace` argument can't have a `null` value.

**Return Value**
- Type: `String`

**getText()**
Returns the text for this node.

**Signature**
```java
public String getText()
```

**Return Value**
- Type: `String`

**insertBefore(newChild, refChild)**
Inserts a new child node before the specified node.

**Signature**
```java
public Dom.XmlNode insertBefore(Dom.XmlNode newChild, Dom.XmlNode refChild)
```

**Parameters**
- `newChild`
  - Type: `Dom.XmlNode`
  - The node to insert.
- `refChild`
  - Type: `Dom.XmlNode`
  - The node before the new node.

**Return Value**
- Type: `Dom.XmlNode`
Usage

- If `refChild` is `null`, `newChild` is inserted at the end of the list.
- If `refChild` doesn’t exist, an exception is thrown.

**removeAttribute(key, keyNamespace)**
Removes the attribute with the given key and key namespace. Returns `true` if successful, `false` otherwise.

**Signature**

```java
public Boolean removeAttribute(String key, String keyNamespace)
```

**Parameters**

- `key` Type: `String`
- `keyNamespace` Type: `String`

**Return Value**
Type: `Boolean`

**removeChild(childNode)**
Removes the given child node.

**Signature**

```java
public Boolean removeChild(Dom.XmlNode childNode)
```

**Parameters**

- `childNode` Type: `Dom.XmlNode`

**Return Value**
Type: `Boolean`

**setAttribute(key, value)**
Sets the key attribute value.

**Signature**

```java
public Void setAttribute(String key, String value)
```
Parameters

key
  Type: String
value
  Type: String

Return Value
Type: Void

setAttributeNs(key, value, keyNamespace, valueNamespace)

Sets the key attribute value.

Signature

public Void setAttributeNs(String key, String value, String keyNamespace, String valueNamespace)

Parameters

key
  Type: String
value
  Type: String
keyNamespace
  Type: String
valueNamespace
  Type: String

Return Value
Type: Void

setNamespace(prefix, namespace)

Sets the namespace for the given prefix.

Signature

public Void setNamespace(String prefix, String namespace)

Parameters

prefix
  Type: String
namespace
  Type: String
Return Value
Type: Void

EventBus Namespace

The EventBus namespace provides classes and methods for platform events and Change Data Capture events. The following are the classes in the EventBus namespace.

IN THIS SECTION:

- ChangeEventHeader Class
  Contains header fields of Change Data Capture events.

- EventPublishFailureCallback Interface
  Implement this interface to track platform event messages that failed to publish. The onFailure() method in this interface is called when the final result of the asynchronous publish operation becomes available.

- EventPublishSuccessCallback Interface
  Implement this interface to track platform event messages that were published successfully. The onSuccess() method in this interface is called when the final result of the asynchronous publish operation becomes available.

- FailureResult Interface
  Contains the result of an Apex publish callback when the event publishing failed. This interface is used as a parameter in the onFailure method of the EventPublishFailureCallback interface.

- SuccessResult Interface
  Contains the result of an Apex publish callback when the event publishing succeeded. This interface is used as a parameter in the onSuccess method of the EventPublishSuccessCallback interface.

- TestBroker Class
  Contains methods that simulate the successful delivery or failed publishing of platform event or change event messages in an Apex test.

- TriggerContext Class
  Provides information about the platform event or change event trigger that’s currently executing, such as how many times the trigger was retried due to the EventBus.RetryableException. Also, provides a method to resume trigger executions.

SEE ALSO:
Platform Events Developer Guide

ChangeEventHeader Class
Contains header fields of Change Data Capture events.

Namespace
EventBus
IN THIS SECTION:

ChangeEventHeader Properties

SEE ALSO:

Change Data Capture Developer Guide

ChangeEventHeader Properties

The following are properties for ChangeEventHeader.

IN THIS SECTION:

changedfields
A list of the fields that were changed in an update operation, including the LastModifiedDate system field. This field is empty for other operations, including record creation. This property is available in Apex saved using API version 47.0 or later.

changeorigin
Only populated for changes done by API apps or from Lightning Experience; empty otherwise. The Salesforce API and the API client ID that initiated the change, if set by the client. Use this field to detect whether your app initiated the change to not process the change again and potentially avoid a deep cycle of changes.

changetype
The operation that caused the change.

commitnumber
The system change number (SCN) of a committed transaction, which increases sequentially. This field is provided for diagnostic purposes. The field value is not guaranteed to be unique in Salesforce—it is unique only in a single database instance. If your Salesforce org migrates to another database instance, the commit number might not be unique or sequential.

committimestamp
The date and time when the change occurred, represented as the number of milliseconds since January 1, 1970 00:00:00 GMT.

commituser
The ID of the user that ran the change operation.

difffields
Contains the names of fields whose values are sent as a unified diff because they contain large text values.

difffields
The API name of the standard or custom object that the change pertains to. For example, Account or MyObject__c.

difffields
Contains the names of fields whose values were changed to null in an update operation. Use this field in Apex change event messages to determine if a field was changed to null in an update and isn’t an unchanged field.

recordids
One or more record IDs for the changed records. Typically, this field contains one record ID. If in one transaction the same change occurred in multiple records of the same object type during one second, Salesforce merges the change notifications. In this case, Salesforce sends one change event for all affected records and the recordIds field contains the IDs for all records that have the same change.

sequencenumber
The sequence of the change within a transaction. The sequence number starts from 1.
transactionkey
A string that uniquely identifies each Salesforce transaction. You can use this key to identify and group all changes that were made in the same transaction.

changedfields
A list of the fields that were changed in an update operation, including the LastModifiedDate system field. This field is empty for other operations, including record creation. This property is available in Apex saved using API version 47.0 or later.

Signature
public List<String> changedfields {get; set;}

Property Value
Type: List<String>

changeorigin
Only populated for changes done by API apps or from Lightning Experience; empty otherwise. The Salesforce API and the API client ID that initiated the change, if set by the client. Use this field to detect whether your app initiated the change to not process the change again and potentially avoid a deep cycle of changes.

Signature
public String changeorigin {get; set;}

Property Value
Type: String
The format of the changeOrigin field value is:
com/salesforce/api/<API_Name>/<API_Version>;client=<Client_ID>

- <API_Name> is the name of the Salesforce API used to make the data change. It can take one of these values: soap, rest, bulkapi, xmlrpc, oldsoap, toolingsoap, toolingrest, apex, apexdebuggerrest.
- <API_Version> is the version of the API call that made the change and is in the format XX.X.
- <Client_ID> is a string that contains the client ID of the app that initiated the change. If the client ID is not set in the API call, client=<Client_ID> is not appended to the changeOrigin field.

Example:
com/salesforce/api/soap/49.0;client=Astro
The client ID is set in the Call Options header of an API call. For an example on how to set the Call Options header, see:
- REST API: Sforce-Call-Options Header. (Bulk API also uses the Sforce-Call-Options header.)
- SOAP API: CallOptions Header. (Apex API also uses the CallOptions element.)

changetype
The operation that caused the change.
**Signature**

```java
public String changeType {get; set;}
```

**Property Value**

Type: `String`

Can be one of the following values:

- `CREATE`
- `UPDATE`
- `DELETE`
- `UNDELETE`

For gap events, the change type starts with the `GAP_` prefix.

- `GAP_CREATE`
- `GAP_UPDATE`
- `GAP_DELETE`
- `GAP_UNDELETE`

For overflow events, the change type is `GAP_OVERFLOW`.

---

**commitnumber**

The system change number (SCN) of a committed transaction, which increases sequentially. This field is provided for diagnostic purposes. The field value is not guaranteed to be unique in Salesforce—it is unique only in a single database instance. If your Salesforce org migrates to another database instance, the commit number might not be unique or sequential.

**Signature**

```java
public Long commitNumber {get; set;}
```

**Property Value**

Type: `Long`

---

**committimestamp**

The date and time when the change occurred, represented as the number of milliseconds since January 1, 1970 00:00:00 GMT.

**Signature**

```java
public Long committimestamp {get; set;}
```

**Property Value**

Type: `Long`

---

**commituser**

The ID of the user that ran the change operation.
**Signature**

public String commituser {get; set;}

**Property Value**

Type: String

**difffields**

Contains the names of fields whose values are sent as a unified diff because they contain large text values.

**Signature**

public List<String> difffields {get; set;}

**Property Value**

Type: List<String>

SEE ALSO:

*Change Data Capture Developer Guide: Sending Data Differences for Fields of Updated Records*

**entityname**

The API name of the standard or custom object that the change pertains to. For example, Account or MyObject__c.

**Signature**

public String entityname {get; set;}

**Property Value**

Type: String

**nulledfields**

Contains the names of fields whose values were changed to null in an update operation. Use this field in Apex change event messages to determine if a field was changed to null in an update and isn't an unchanged field.

**Signature**

public List<String> nulledfields {get; set;}

**Property Value**

Type: List<String>
**recordids**

One or more record IDs for the changed records. Typically, this field contains one record ID. If in one transaction the same change occurred in multiple records of the same object type during one second, Salesforce merges the change notifications. In this case, Salesforce sends one change event for all affected records and the `recordIds` field contains the IDs for all records that have the same change.

**Signature**

```java
public List<String> recordids {get; set;}
```

**Property Value**

Type: `List<String>`

Examples of operations with same changes are:

- Update of `fieldA` to `valueA` in `Account` records.
- Deletion of `Account` records.
- Renaming or replacing a picklist value that results in updating the field value in all affected records.

The `recordIds` field can contain a wildcard value when a change event message is generated for custom field type conversions that cause data loss. In this case, the `recordIds` value is the three-character prefix of the object, followed by the wildcard character `*`. For example, for accounts, the value is `001*`.

**sequencenumber**

The sequence of the change within a transaction. The sequence number starts from 1.

**Signature**

```java
public Integer sequencenumber {get; set;}
```

**Property Value**

Type: `Integer`

A lead conversion is an example of a transaction that can have multiple changes. A lead conversion results in the following sequence of changes, all within the same transaction.

1. Create an account
2. Create a contact
3. Create an opportunity
4. Update a lead

**transactionkey**

A string that uniquely identifies each Salesforce transaction. You can use this key to identify and group all changes that were made in the same transaction.

**Signature**

```java
public String transactionkey {get; set;}
```
Property Value
Type: String

EventPublishFailureCallback Interface
Implement this interface to track platform event messages that failed to publish. The `onFailure()` method in this interface is called when the final result of the asynchronous publish operation becomes available.

Namespace
EventBus

Usage
For more information, see <link>Get the Result of Asynchronous Platform Event Publishing with Apex Publish Callbacks</link> in the Platform Events Developer Guide.

IN THIS SECTION:
- EventPublishFailureCallback Methods
- EventPublishFailureCallback Example Implementation

EventPublishFailureCallback Methods
The following are methods for EventPublishFailureCallback.

IN THIS SECTION:
- onFailure(result)
  The system invokes this method when the final result of `EventBus.publish` is available and the publishing of the platform event message failed.

**onFailure(result)**
The system invokes this method when the final result of `EventBus.publish` is available and the publishing of the platform event message failed.

**Signature**
`public void onFailure(eventbus.FailureResult result)`

**Parameters**
- `result`
  Type: `EventBus.FailureResult`
  The final result of `EventBus.publish`.
Return Value
Type: void

EventPublishFailureCallback Example Implementation
For an example implementation and a test class, see <link>Get the Result of Asynchronous Platform Event Publishing with Apex Publish Callbacks</link> in the Platform Events Developer Guide.

EventPublishSuccessCallback Interface
Implement this interface to track platform event messages that were published successfully. The `onSuccess()` method in this interface is called when the final result of the asynchronous publish operation becomes available.

Namespace
EventBus

Usage
For more information, see <link>Get the Result of Asynchronous Platform Event Publishing with Apex Publish Callbacks</link> in the Platform Events Developer Guide.

IN THIS SECTION:
- EventPublishSuccessCallback Methods
- EventPublishSuccessCallback Example Implementation

EventPublishSuccessCallback Methods
The following are methods for EventPublishSuccessCallback.

IN THIS SECTION:
- `onSuccess(result)`
  
  The system invokes this method when the final result of `EventBus.publish` is available and the publishing of the platform event message succeeded.

`onSuccess(result)`

The system invokes this method when the final result of `EventBus.publish` is available and the publishing of the platform event message succeeded.

Signature

```java
public void onSuccess(eventbus.SuccessResult result)
```
Parameters

result
Type: EventBus.SuccessResult
The final result of EventBus.publish.

Return Value
Type: void

EventPublishSuccessCallback Example Implementation
For an example implementation and a test class, see <link>Get the Result of Asynchronous Platform Event Publishing with Apex Publish Callbacks</link> in the Platform Events Developer Guide.

FailureResult Interface
Contains the result of an Apex publish callback when the event publishing failed. This interface is used as a parameter in the onFailure method of the EventPublishFailureCallback interface.

Namespace
EventBus

IN THIS SECTION:
FailureResult Methods

FailureResult Methods
The following are methods for FailureResult.

IN THIS SECTION:
getEventUuids() Returns a list of EventUuid field values of each platform event that is included in EventBus.EventPublishFailureCallback.

getEventUuids() Returns a list of EventUuid field values of each platform event that is included in EventBus.EventPublishFailureCallback.

Signature
public List<String> getEventUuids()
Return Value
Type: List<String>

SuccessResult Interface
Contains the result of an Apex publish callback when the event publishing succeeded. This interface is used as a parameter in the onSuccess method of the EventBus.EventPublishSuccessCallback interface.

Namespace
EventBus

IN THIS SECTION:
SuccessResult Methods

SuccessResult Methods
The following are methods for SuccessResult.

IN THIS SECTION:
getEventUuids()

Returns a list of EventUuid field values of each platform event that is included in the EventBus.EventPublishSuccessCallback.

getEventUuids()
Returns a list of EventUuid field values of each platform event that is included in the EventBus.EventPublishSuccessCallback.

Signature
public List<String> getEventUuids()

Return Value
Type: List<String>

TestBroker Class
Contains methods that simulate the successful delivery or failed publishing of platform event or change event messages in an Apex test.

Namespace
EventBus
TestBroker Methods

The following are methods for TestBroker.

deliver()
Delivers platform event messages to the test event bus. Use this method to deliver test event messages multiple times and verify that event subscribers have processed the test events each step of the way.

fail()
Causes the publishing of platform event messages to fail in the test event bus. Use this method to test Apex publish callbacks.

**deliver()**

Delivers platform event messages to the test event bus. Use this method to deliver test event messages multiple times and verify that event subscribers have processed the test events each step of the way.

**Signature**

```
public void deliver()
```

**Return Value**

Type: void

**Usage**

Enclose `Test.getEventBus().deliver()` within the `Test.startTest()` and `Test.stopTest()` statement block.

```java
Test.startTest();
// Create test events
// ...
// Publish test events with EventBus.publish()
// ...
// Deliver test events
Test.getEventBus().deliver();
// Perform validation
// ...
Test.stopTest();
```

**SEE ALSO:**

*Platform Events Developer Guide*

**fail()**

Causes the publishing of platform event messages to fail in the test event bus. Use this method to test Apex publish callbacks.
Signature

public void fail()

Return Value

Type: void

Usage

// Create test events
// ...
// Publish test events with EventBus.publish()
// ...
// Fail publishing of test events
Test.getEventBus().fail();
// Perform validation
// ...

For more information, see <link>Get the Result of Asynchronous Platform Event Publishing with Apex Publish Callbacks</link> in the Platform Events Developer Guide.

TriggerContext Class

Provides information about the platform event or change event trigger that’s currently executing, such as how many times the trigger was retried due to the EventBus.RetryableException. Also, provides a method to resume trigger executions.

Namespace

EventBus

IN THIS SECTION:

  TriggerContext Properties
  TriggerContext Methods

TriggerContext Properties

The following are properties for TriggerContext.

IN THIS SECTION:

  lastError
Read-only. The error message that the last thrown EventBus.RetryableException contains.

  retries
Read-only. The number of times the trigger was retried due to throwing the EventBus.RetryableException.
**lastError**

Read-only. The error message that the last thrown `EventBus.RetryableException` contains.

**Signature**

```
public String lastError {get;}
```

**Property Value**

Type: `String`

**Usage**

The error message that this property returns is the message that was passed in when creating the `EventBus.RetryableException` exception, as follows.

```java
throw new EventBus.RetryableException(
    'Condition is not met, so retrying the trigger again.');
```

**retries**

Read-only. The number of times the trigger was retried due to throwing the `EventBus.RetryableException`.

**Signature**

```
public Integer retries {get;}
```

**Property Value**

Type: `Integer`

**TriggerContext Methods**

The following are methods for `TriggerContext`.

**IN THIS SECTION:**

- `currentContext()`
  Returns an instance of the `EventBus.TriggerContext` class containing information about the currently executing trigger.

- `getResumeCheckpoint()`
  Returns the replay ID that was set by `setResumeCheckpoint()`. The returned value is the replay ID of the event message after which trigger processing resumes in a new trigger invocation.

- `setResumeCheckpoint(resumeReplayId)`
  Sets a checkpoint in the event stream where the platform event trigger resumes execution in a new invocation. Use this method to recover from limit and uncaught exceptions, or to control the number of events processed in one trigger execution. When calling this method, pass in the replay ID of the last successfully processed event message. When the trigger stops execution before all events in `Trigger.New` are processed, either because of an uncaught exception or intentionally, the trigger is invoked again. The new execution starts with the event message in the stream after the one with the checkpointed Replay ID.
currentContext()
Returns an instance of the EventBus.TriggerContext class containing information about the currently executing trigger.

Signature
public static eventbus.TriggerContext currentContext()

Return Value
Type: EventBus.TriggerContext
Information about the currently executing trigger.

getResumeCheckpoint()
Returns the replay ID that was set by setResumeCheckpoint(). The returned value is the replay ID of the event message after which trigger processing resumes in a new trigger invocation.

Signature
public String getResumeCheckpoint()

Return Value
Type: String

setResumeCheckpoint(resumeReplayId)
Sets a checkpoint in the event stream where the platform event trigger resumes execution in a new invocation. Use this method to recover from limit and uncaught exceptions, or to control the number of events processed in one trigger execution. When calling this method, pass in the replay ID of the last successfully processed event message. When the trigger stops execution before all events in Trigger.New are processed, either because of an uncaught exception or intentionally, the trigger is invoked again. The new execution starts with the event message in the stream after the one with the checkpointed Replay ID.

Signature
public void setResumeCheckpoint(String resumeReplayId)

Parameters
resumeReplayId
Type: String
The replay ID of the last successfully processed platform event message, after which to resume processing in a new trigger execution context.

Return Value
Type: void
Usage

The method throws an EventBus.InvalidReplayIdException if the supplied Replay ID is not valid—the replay ID is not in the current trigger batch of events, in the Trigger.new list.

Example

This snippet shows how to call the method and pass in the replayId property of an event instance.

```java
EventBus.TriggerContext.currentContext().setResumeCheckpoint(event.replayId);
```

ExternalService Namespace

The ExternalService namespace provides dynamically generated Apex service interfaces and Apex classes for complex object data types.

The ExternalService namespace doesn’t define a fixed set of classes. The namespace reflects OpenAPI-compatible external service registrations with active operations for type-safe outbound calls. The object schema, in the API specification that is associated with the registered external service, maps to Apex types.

SEE ALSO:

* Salesforce Help: Invoke External Service Callouts Using Apex*

Flow Namespace

The Flow namespace provides a class for advanced Visualforce controller access to flows.

The following is the class in the Flow namespace.

IN THIS SECTION:

**Interview Class**

The Flow.Interview class provides advanced controller access to flows and the ability to start a flow.

Interview Class

The Flow.Interview class provides advanced controller access to flows and the ability to start a flow.

Namespace

Flow

Usage

SOQL and DML limits apply during flow execution. See Per-Transaction Flow Limits in the Salesforce Help.

To create an Interview object, you have two options.

- Create the object directly in your class by using:
- No namespace: Flow.Interview.flowName
- Namespace: Flow.Interview.namespace.flowName

Create the object dynamically by using createInterview()

Note: We recommend only using createInterview() if you need to reuse your method or class. Using createInterview() has these drawbacks.

- If you package a class that uses createInterview(), you have to add the associated flow manually.
- If you delete a flow, Salesforce doesn’t check if it’s referenced with createInterview().

Examples: Starting Flow Interviews

The following examples are all sample controllers that start an interview for the flow from the Build a Discount Calculator project on Trailhead. Each shows a different permutation, based on:

- Whether the interview is created statically, with Flow.Interview.myFlow, or dynamically, with createInterview().
- Whether the flow is managed or local.

### Interview Created Statically for a Local Flow

```java
{  
  Map<String, Object> inputs = new Map<String, Object>();  
  inputs.put('AccountID', myAccount);  
  inputs.put('OpportunityID', myOppty);  

  Flow.Interview.Calculate_discounts myFlow =  
  new Flow.Interview.Calculate_discounts(inputs);  
  myFlow.start();
}
```

### Interview Created Dynamically for a Local Flow

```java
public void callFlow(String flowName, Map<String, Object> inputs) {
  Flow.Interview myFlow = Flow.Interview.createInterview(flowName, inputs);
  myFlow.start();
}
```

### Interview Created Statically for a Managed Flow

```java
{  
  Map<String, Object> inputs = new Map<String, Object>();  
  inputs.put('AccountID', myAccount);  
  inputs.put('OpportunityID', myOppty);  

  Flow.Interview.myNamespace.Calculate_discounts myFlow =  
  new Flow.Interview.myNamespace.Calculate_discounts(inputs);  
  myFlow.start();
}
```

### Interview Created Dynamically for a Managed Flow

```java
public void callFlow(String namespace, String flowName, Map<String, Object> inputs) {
  Flow.Interview myFlow = Flow.Interview.createInterview(namespace, flowName, inputs);
  myFlow.start();
}
```
Example: Getting Variable Values

This sample uses the `getVariableValue` method to obtain breadcrumb (navigation) information from a flow. If that flow contains subflow elements, and each of the referenced flows also contains a `vaBreadCrumb` variable, you can provide users with breadcrumbs regardless of which flow the interview is running.

```java
public class SampleController {

    // Instance of the flow
    public Flow.Interview.Flow_Template_Gallery myFlow {get; set;}

    public String getBreadCrumb() {
        String aBreadCrumb;
        if (myFlow==null) { return 'Home';}
        else aBreadCrumb = (String) myFlow.getVariableValue('vaBreadCrumb');

        return(aBreadCrumb==null ? 'Home': aBreadCrumb);
    }
}
```

Interview Methods

The following are instance methods for `Interview`.

`createInterview(namespace, flowName, inputVariables)`

Creates an interview for a namespaced flow.

**Signature**

```java
public static Flow.Interview createInterview(String namespace, String flowName, Map<String,ANY> inputVariables)
```

**Parameters**

`namespace`

Type: `String`

The flow's namespace.

`flowName`

Type: `String`

The flow's API name.

`inputVariables`

Type: `Map<String, Object>`

Initial values for the flow's input variables.

**Return Value**

Type: `Flow.Interview`
Usage

Use this method to dynamically create a Flow.Interview object for the `start()` method.

How you get output variable values from an interview depends on the type of the Apex variable where you’re storing the interview.

- If the variable is cast to a specific flow, you can use `myFlow.myVar` to access a variable, where `myVar` is the name of the variable.
  ```java
  system.debug('My Output Variable: ' + myFlow.varName);
  ```

- If the variable is of type Flow.Interview but not cast to a specific flow, you must use `getVariableValue()` to access the flow's variables.
  ```java
  system.debug('My Output Variable: ' + myFlow.getVariableValue('varName'));
  ```

If the flow doesn’t exist in the current org, a `TypeException` is thrown.

`createInterview(flowName, inputVariables)`

Creates an interview for a flow.

**Signature**

```java
public static Flow.Interview createInterview(String flowName, Map<String,Object> inputVariables)
```

**Parameters**

- `flowName`
  
  **Type:** `String`
  
  The flow’s API name.

- `inputVariables`
  
  **Type:** `Map<String,Object>`
  
  Initial values for the flow’s input variables.

**Return Value**

**Type:** `Flow.Interview`

**Usage**

Use this method to dynamically create a Flow.Interview object for the `start()` method.

How you get output variable values from an interview depends on the type of the Apex variable where you’re storing the interview.

- If the variable is cast to a specific flow, you can use `myFlow.myVar` to access a variable, where `myVar` is the name of the variable.
  ```java
  system.debug('My Output Variable: ' + myFlow.varName);
  ```

- If the variable is of type Flow.Interview but not cast to a specific flow, you must use `getVariableValue()` to access the flow's variables.
  ```java
  system.debug('My Output Variable: ' + myFlow.getVariableValue('varName'));
  ```

If the flow doesn’t exist in the current org, a `TypeException` is thrown.
getVariableValue (variableName)
Returns the value of the specified flow variable. The flow variable can be in the flow embedded in the Visualforce page, or in a separate flow that is called by a subflow element.

Signature
public Object getVariableValue(String variableName)

Parameters
variableName
Type: String
Specifies the unique name of the flow variable.

Return Value
Type: Object

Usage
The returned variable value comes from whichever flow the interview is running. If the specified variable can’t be found in that flow, the method returns null.

This method checks for the existence of the variable at run time only, not at compile time.

start()
Starts an instance (interview) for an autolaunched or user provisioning flow.

Signature
public Void start()

Return Value
Type: Void

Usage
This method can be used only with flows that have one of these types.

• Autolaunched Flow
• User Provisioning Flow

For details, see “Flow Types” in Salesforce Help.

When a flow user invokes an autolaunched flow, the active flow version runs. If there's no active version, the latest version runs. When a flow admin invokes a flow, the latest version always runs.
Functions Namespace

The Functions namespace provides classes and methods used to invoke and manage Salesforce Functions. Salesforce Functions is your code, run on demand, in the Salesforce Functions trusted elastic compute cloud. Upload your complex business logic code, written using your preferred languages and frameworks, and Salesforce Functions takes care of everything else necessary to invoke your code in a secure, multi-tenant aware, and self-scaling environment. For more details on Salesforce Functions, see Salesforce Functions.

The following are the classes in the functions namespace.

IN THIS SECTION:

- **Function Class**
  Use the Function class to access deployed Salesforce Functions, and invoke them synchronously or asynchronously.

- **FunctionCallback Interface**
  Represents the callback Salesforce calls when an asynchronous, queued Function invocation has completed.

- **FunctionErrorType Enum**
  Represents the error type of FunctionInvocationError.

- **FunctionInvocation Interface**
  Use FunctionInvocation to get the status and results of a synchronous or asynchronous Function invocation.

- **FunctionInvocationError Interface**
  Use FunctionInvocationError to get detailed error information about a failed Function invocation.

- **FunctionInvocationStatus Enum**
  Represents the status of a Function invocation.

- **FunctionInvokeMock Interface**
  Use the FunctionInvokeMock interface to mock Salesforce Functions responses during testing.

- **MockFunctionInvocationFactory Class**
  Use the MockFunctionInvocationFactory methods to generate appropriate mock responses for testing Salesforce Functions.

Function Class

Use the Function class to access deployed Salesforce Functions, and invoke them synchronously or asynchronously.

Namespace

functions

Usage

The Function class represents an instance of a deployed Function you can invoke from your org. You can invoke Functions synchronously, or asynchronously using asynchronous Apex.

If your Function takes longer than 2 minutes to return, the request will time out. To avoid timing out, consider using asynchronous invocation. Invoking a Function asynchronously doesn’t count against asynchronous Apex limits, such as Apex Queueable limits.
Before synchronously invoking a Function, commit any pending data operations in Apex, otherwise you will get a CalloutException. For asynchronous invocations, the queued invocation won’t happen if the Apex transaction is not committed. Any data operations that happen in the Function itself are not considered part of the Apex transaction.

Functions cannot be invoked in an Apex test. A “Function invocations from Apex tests are not supported” CalloutException is thrown if Apex determines that a Function is being invoked during a test. If you must run tests against code that invokes Functions you’ll need to mock your Function invocations during the tests. See FunctionInvocation Example Implementation for an example of a mocked FunctionInvocation that you can use in testing.

**Example**

The following example synchronously invokes a deployed “accountfunction” Function:

```java
functions.Function accountFunction = functions.Function.get('MyProject.accountfunction');
functions.FunctionInvocation invocation = accountFunction.invoke('{ "accountName" : "Acct",  
  "contactName" : "MyContact", "opportunityName" : "Oppty" }');
String jsonResponse = invocation.getResponse();
```

The following example asynchronously invokes a deployed “AccountFunction” Function, using the provided callback:

```java
functions.Function accountFunction = functions.Function.get('MyProject.accountfunction');
accountFunction.invoke('{ "accountName" : "Acct", "contactName" : "MyContact",  
  "opportunityName" : "Oppty" }', new MyCallback());

public class MyCallback  
  implements functions.FunctionCallback { 
    public void handleResponse(functions.FunctionInvocation result) {
    // Handle result of function invocation
    // ...
    }
  }
```

**IN THIS SECTION:**

  Function Methods

**Function Methods**

The following are methods for `Function`.

**IN THIS SECTION:**

  `get(functionName)`
  Returns the Function instance for the named Function and Project. The Function must be properly deployed and have appropriate permissions to work with the org running your Apex code.

  `get(namespace, functionName)`
  Returns the Function instance for the named Function, Project, and Namespace. The Function must be properly deployed and have appropriate permissions to work with the org running your Apex code.

  `invoke(payload, callback)`
  Invokes the Function asynchronously.
invoke(payload)
Invokes the Function synchronously.

get(functionName)
Returns the Function instance for the named Function and Project. The Function must be properly deployed and have appropriate permissions to work with the org running your Apex code.

**Signature**

```java
public static functions.Function get(String functionName)
```

**Parameters**

- **functionName**
  - Type: String
  - The name of the Salesforce Function and the Functions Project that the Function is part of. The format of the parameter string is "project name.function name". For example, to retrieve the `generatepdf` Function in the `Onboarding` Function Project, use `Onboarding.generatepdf`. The Function and Project must be deployed to a compute environment connected to the org.

**Return Value**

- Type: functions.Function
  - Returns a Function instance that you can invoke.

**Usage**

The `Function.get()` method can throw the following exceptions:

- **InvalidParameterValueException** — The `functionName` parameter doesn’t have the correct `project name.function name` format.
- **NoDataFoundException** — The project or Function name provided in the `functionName` parameter couldn’t be found. Make sure the project and Function name are spelled correctly and that the project and Function have been properly deployed.

get(namespace, functionName)
Returns the Function instance for the named Function, Project, and Namespace. The Function must be properly deployed and have appropriate permissions to work with the org running your Apex code.

**Signature**

```java
public static functions.Function get(String namespace, String functionName)
```

**Parameters**

- **namespace**
  - Type: String
  - The name of the Namespace that both the Salesforce Function and the Functions Project are part of. The org the Function is in must be global to access across namespaces. Default value is the same org where the method is being called.
**Function Name**

Type: String

The name of the Salesforce Function and the Functions Project that the Function is part of. The format of the parameter string is "**project name.function name**". For example, to retrieve the `generatepdf` Function in the `Onboarding` Function Project, use `Onboarding.generatepdf`. The Function and Project must be deployed to a compute environment connected to the org.

**Return Value**

Type: functions.Function

Returns a Function instance that you can invoke.

**Usage**

The `Function.get()` method can throw the following exceptions:

- **InvalidParameterValueException** — The `functionName` parameter doesn’t have the correct **project name.function name** format.
- **NoDataFoundException** — The project or Function name provided in the `functionName` parameter couldn’t be found. Make sure the project and Function name are spelled correctly and that the project and Function have been properly deployed.
- **RuntimeException** — The function is **public** yet references a function across namespaces. Make sure to retrieve references across namespaces only in a global org.

**invoke(payload, callback)**

Invokes the Function asynchronously.

**Signature**

```java
public functions.FunctionInvocation invoke(String payload, functions.FunctionCallback callback)
```

**Parameters**

- **payload**
  
  Type: String
  
  The payload data that gets passed to the Function. Specify your payload data in a JSON-format string.

- **callback**
  
  Type: functions.FunctionCallback
  
  A FunctionCallback implementation that gets called when your Function is invoked asynchronously.

**Return Value**

Type: functions.FunctionInvocation

Returns a FunctionInvocation that contains information about the results of the invocation, such as the Function response, or error results.
Usage

The `Function.invoke(payload, callback)` method can throw the following exceptions:

- **CalloutException** — One of the following conditions causes this exception to be thrown:
  - Salesforce Functions isn’t enabled on the current org. For more details on enabling Functions, see [Configure Orgs for Functions](#) in the Functions Developer Guide.
  - The Function is being invoked in an Apex test. Functions can’t be invoked in tests.
  - The “Functions” permission set is missing or has incorrect permissions for `FunctionInvocationRequest`. For more details on the correct permissions for `FunctionInvocationRequest` see [Function Permissions](#) in the Functions Developer Guide.
  - The provided payload isn’t valid JSON.
  - The Function hasn’t completed deployment to a compute environment or invocation request returns a 404 HTTP error.

- **InvalidParameterValueException** — The `callback` parameter is null or references a class that doesn’t implement `functions.FunctionCallback`.

- **NoDataFoundException** — A reference for the Function couldn’t be found in the current org. Make sure the project and Function have been properly deployed.

### `invoke(payload)`

Invokes the Function synchronously.

### Signature

```java
public functions.FunctionInvocation invoke(String payload)
```

### Parameters

- **payload**
  - Type: `String`
  - The payload data that gets passed to the Function. Specify your payload data in a JSON-format string.

### Return Value

Type: `functions.FunctionInvocation`  
Returns a `FunctionInvocation` that contains information about the results of the invocation, such as the Function response, or error results.

Usage

The `Function.invoke(payload)` method can throw the following exceptions:

- **CalloutException** — One of the following conditions causes this exception to be thrown:
  - Salesforce Functions isn’t enabled on the current org. For more details on enabling Functions, see [Configure Orgs for Functions](#) in the Functions Developer Guide.
  - The Function is being invoked in an Apex test. Functions can’t be invoked in tests.
  - The provided payload isn’t valid JSON.
  - There are pending DML operations.
The Function is being synchronously invoked from an Apex trigger.
The Function hasn’t completed deployment to a compute environment or invocation request returns a 404 HTTP error.
The Function request returns a 5xx HTTP error.
The Function invocation has exceeded the time limit for synchronous invocations. For details on the time limit and work-arounds, see Limits in the Functions Developer Guide.

- NoDataFoundException — A reference for the Function couldn’t be found in the current org. Make sure the project and Function have been properly deployed.

FunctionCallback Interface

Represents the callback Salesforce calls when an asynchronous, queued Function invocation has completed.

Namespace

functions

Usage

When invoking Functions asynchronously via `Function.invoke(payload, callback)`, you provide your own class that implements FunctionCallback.

IN THIS SECTION:
- FunctionCallback Methods
- FunctionCallback Example Implementation

FunctionCallback Methods

The following are methods for FunctionCallback.

IN THIS SECTION:
- `handleResponse(var1)`
  Called when an asynchronous Function invocation has completed.

**handleResponse (var1)**

Called when an asynchronous Function invocation has completed.

**Signature**

```java
public void handleResponse(functions.FunctionInvocation var1)
```

**Parameters**

- `var1`
  Type: `functions.FunctionInvocation`
The result parameter contains JSON response information and error information.

**Return Value**
Type: void

**FunctionCallback Example Implementation**
This is an example implementation of the functions.FunctionCallback interface.

```java
public class MyCallback
   implements functions.FunctionCallback {
   public void handleResponse(functions.FunctionInvocation result) {
      // Handle result of function invocation
      String jsonResponse = result.getResponse();
      System.debug('Got response ' + jsonResponse);
      JSONParser parser = JSON.createParser(jsonResponse);
      // Process JSON using your own data class...
   }
}
```

The following example uses this implementation when invoking a Function asynchronously:

```java
myFunction.invoke('{ "accountName" : "Acct", "contactName" : "MyContact", "opportunityName" : "Oppty" }', new MyCallback());
```

**FunctionErrorType Enum**
Represents the error type of FunctionInvocationError.

**Enum Values**
These are the values of the functions.FunctionErrorType enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FUNCTION_EXCEPTION</td>
<td>A known exception resulting from the Function logic itself. Examples include an exception thrown from the Function code, or an exception thrown from a library or framework the Function uses.</td>
</tr>
<tr>
<td>RUNTIME_EXCEPTION</td>
<td>A known exception resulting from the Salesforce Functions runtime. For example, a malformed payload passed to the Function when invoked results in this error type.</td>
</tr>
<tr>
<td>UNEXPECTED_FUNCTION_EXCEPTION</td>
<td>An unknown exception. For example, a network or system-level issue within the Salesforce Functions infrastructure results in this error type.</td>
</tr>
</tbody>
</table>

**FunctionInvocation Interface**
Use FunctionInvocation to get the status and results of a synchronous or asynchronous Function invocation.
Namespace
functions

Usage
The results of a Function invocation are passed back via FunctionInvocation. Use this instance to determine if the invocation was successful, and any results from the Function invocation.

You can also implement your own FunctionInvocation interface if you run Apex tests with your Function invocation code. Your test code can create and use your own FunctionInvocation instance in place of using the results from a call to Function.invoke().

IN THIS SECTION:
  FunctionInvocation Methods
  FunctionInvocation Example Implementation

FunctionInvocation Methods
The following are methods for FunctionInvocation.

IN THIS SECTION:
  getError()
  getInvocationId()
  getResponse()
  getStatus()

getError()
Returns error information for a Function invocation.

Signature
public functions.FunctionInvocationError getError()

Return Value
Type: functions.FunctionInvocationError
Contains a FunctionInvocationError instance that you can use to get information about any invocation errors. If the Function was invoked successfully, the returned instance is null.

getInvocationId()
Returns the invocation ID of the Function invocation.
**Signature**

```java
public String getInvocationId()
```

**Return Value**

Type: `String`

This ID is available after a call to either the synchronous or asynchronous `Function.invoke()` methods. For asynchronous invocations, this ID can be used to check the status of the queued invocation.

**getResponse()**

Returns the response string of the Function invocation.

**Signature**

```java
public String getResponse()
```

**Return Value**

Type: `String`

The response string is the raw request JSON response, which can be parsed using the `JSONParser` Class.

**getStatus()**

Returns the status of the Function invocation.

**Signature**

```java
public functions.FunctionInvocationStatus getStatus()
```

**Return Value**

Type: `functions.FunctionInvocationStatus`

The result of the invocation, such as `FunctionInvocationStatus.SUCCESS` or `FunctionInvocationStatus.ERROR`.

**FunctionInvocation Example Implementation**

This is an example implementation of the `functions.FunctionInvocation` interface.

```java
public class MyFunctionInvocationError implements functions.FunctionInvocationError {
    public String getMessage() {
        return 'Mock error message for testing';
    }
    public functions.FunctionErrorType getType() {
        return functions.FunctionErrorType.FUNCTION_EXCEPTION;
    }
}

public class MyFunctionInvocation
```
```java
implements functions.FunctionInvocation {
    public functions.FunctionInvocationStatus getStatus() {
        return functions.FunctionInvocationStatus.ERROR;
    }
    public String getResponse() {
        return 'Mock response for testing';
    }
    public String getInvocationId() {
        return 'MOCKTESTID';
    }
    public functions.FunctionInvocationError getError() {
        functions.FunctionInvocationError testError = new MyFunctionInvocationError();
        return testError;
    }
}
```

The following example tests the implementation:

```java
functions.FunctionInvocation testInvocation = new MyFunctionInvocation();
if (testInvocation.getStatus() == functions.FunctionInvocationStatus.ERROR) {
    System.debug('Error: ' + (testInvocation.getError() != null ? testInvocation.getError().getMessage() : 'UNKNOWN'));
    return;
}
```

**FunctionInvocationError Interface**

Use FunctionInvocationError to get detailed error information about a failed Function invocation.

**Namespace**

`functions`

**Usage**

FunctionInvocationError contains various error information such as the error message at the time of the error.

**IN THIS SECTION:**

- FunctionInvocationError Methods
- FunctionInvocationError Example Implementation

**FunctionInvocationError Methods**

The following are methods for `FunctionInvocationError`.

**IN THIS SECTION:**

- `getMessage()`
  Returns the error message from a Function invocation error.
**getMessage()**
Returns the error message from a Function invocation error.

**Signature**
```java
public String getMessage()
```

**Return Value**
Type: String

**getType()**
Returns the error type for FunctionInvocationError.

**Signature**
```java
public functions.FunctionErrorType getType()
```

**Return Value**
Type: functions.FunctionErrorType

### FunctionInvocationError Example Implementation
This is an example implementation of the `functions.FunctionInvocationError` interface.

```java
public class MyFunctionInvocationError implements functions.FunctionInvocationError {
    public String getMessage() {
        return 'Mock error message for testing';
    }
    public functions.FunctionErrorType getType() {
        return functions.FunctionErrorType.FUNCTION_EXCEPTION;
    }
}
```

This example tests the implementation.

```java
functions.FunctionInvocationError testError = new MyFunctionInvocationError();
System.debug('Error: ' + testError.getMessage() + ' Type: ' + testError.getType());
```

### FunctionInvocationStatus Enum
Represents the status of a Function invocation.
Enum Values

The following are the values of the `functions.FunctionInvocationStatus` enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ERROR</td>
<td>The invocation failed. Check the <code>FunctionInvocation</code> and <code>FunctionInvocationError</code> returned by the invoke call to debug the issue.</td>
</tr>
<tr>
<td>PENDING</td>
<td>The invocation is pending. If the Function is being invoked asynchronously, the invocation is still in the asynch queue.</td>
</tr>
<tr>
<td>SUCCESS</td>
<td>The invocation succeeded. Use <code>FunctionInvocation.getResponse()</code> with the <code>FunctionInvocation</code> instance returned by the invoke call to get any response returned by the Function.</td>
</tr>
</tbody>
</table>

FunctionInvokeMock Interface

Use the `FunctionInvokeMock` interface to mock Salesforce Functions responses during testing.

Namespace

`functions`

Usage

To mock Salesforce Functions testing, implement an appropriate mock response in the `respond(functionName, payload)` method of the `FunctionInvokeMock` interface. During mock testing of a Salesforce Functions, Apex runtime sends the response specified in the `respond()` method, rather than invoking the function itself. Appropriate success and error messages can be configured with the `createSuccessResponse(invocationId, message)` and `createErrorResponse(invocationId, functionsErrorType, errorMsg)` methods in `Functions.MockFunctionInvocationFactory`.

IN THIS SECTION:
- FunctionInvokeMock Methods
- FunctionInvokeMock Example Implementation

FunctionInvokeMock Methods

The following are methods for `FunctionInvokeMock`.

IN THIS SECTION:
- `respond(functionName, payload)`
  The mock response implemented in the `functions.FunctionInvokeMock` interface. The response is sent by Apex runtime when the `Test.setMock()` method is called.
respond(functionName, payload)

The mock response implemented in the Functions.FunctionInvokeMock interface. The response is sent by Apex runtime when the Test.setMock() method is called.

**Signature**

```
public functions.FunctionInvocation respond(String functionName, String payload)
```

**Parameters**

- **functionName**
  - Type: String
  - The name of the Salesforce Function and the Functions Project that the Function is part of. The format of the parameter string is "project name.function name".

- **payload**
  - Type: String
  - The JSON-format payload data that is passed to the Function.

**Return Value**

Type: FunctionInvocation Interface

The result of the mock call to Salesforce Functions. Appropriate responses can be generated by using the createSuccessResponse() and createErrorResponse() methods in the Functions.MockFunctionInvocationFactory class.

**FunctionInvokeMock Example Implementation**

This is sample implementation of the functions.FunctionInvokeMock interface.

```java
@isTest
public class FunctionsInvokeMockImpl implements functions.FunctionInvokeMock {
    public functions.FunctionInvocation respond(String functionName, String payload) {
        // return mock success response
        String invocationId = '00000000000000000000000000000000';
        String response = 'mockResponse';
        return functions.MockFunctionInvocationFactory.createSuccessResponse(invocationId, response);
    }
}
```

This example shows the minimal setup required for testing synchronous and asynchronous functions and is simplified to include both function invocations and the FunctionCallback class.

```java
@isTest
public class FunctionTest {
    @isTest
    static void testSyncFunctionCall() {
        // Set mock class to respond to function invocations
        Test.setMock( functions.FunctionInvokeMock.class, new FunctionsInvokeMockInner() );
    }
}
```
functions.Function mockedFunction = functions.Function.get('example.function');

Test.startTest();
// Synchronous function call
functions.FunctionInvocation invokeResult = mockedFunction.invoke('{}');
Test.stopTest();

// Verify that the received response contains expected mock values
System.assertEquals(functions.FunctionInvocationStatus.SUCCESS,
                      invokeResult.getStatus());
System.assertEquals('mockResponse', invokeResult.getResponse());
System.assertEquals('000000000000000', invokeResult.getInvocationId());


@Test
static void testAsyncFunctionCall() {
// Set mock class to respond to function invocations
Test.setMock( functions.FunctionInvokeMock.class, new
FunctionsInvokeMockInner());

functions.Function mockedFunction = functions.Function.get('example.function2');

Test.startTest();
// Asynchronous function invocation with callback
mockedFunction.invoke('{}', new DemoCallback());
Test.stopTest();
// Include assertions here about the expected callback processing
}

public class DemoCallback implements functions.FunctionCallback {
public void handleResponse(functions.FunctionInvocation invokeResult) {
  // Handle result of function invocation
  // The callback is included in the example here for convenience
  // It would normally be defined in the classes being tested

  // Verify that the received response contains expected mock values
  System.assertEquals(invokeResult.getStatus(),
                      functions.FunctionInvocationStatus.ERROR);
  functions.FunctionInvocationError resultError = invokeResult.getError();
  System.assertEquals('bang', invokeResult.getError().getMessage());
  System.assertEquals('000000000000000', invokeResult.getInvocationId());
}
}

public class FunctionsInvokeMockInner implements functions.FunctionInvokeMock {
public functions.FunctionInvocation respond(String functionName, String payload) {
  // return mock success response
String invocationId = '000000000000000';

if(functionName == 'example.function2') {
    return functions.MockFunctionInvocationFactory.createErrorResponse(
        invocationId,
        functions.FunctionErrorType.FUNCTION_EXCEPTION,
        'bang');
}

String response = 'mockResponse';
return functions.MockFunctionInvocationFactory.createSuccessResponse(invocationId, response);

MockFunctionInvocationFactory Class

Use the MockFunctionInvocationFactory methods to generate appropriate mock responses for testing Salesforce Functions.

Namespace

functions

Usage

To mock Salesforce Functions testing, implement an appropriate mock response in the respond(functionName,payload) method of the FunctionInvokeMock interface. During mock testing of a Salesforce Functions, the Apex runtime sends the response specified in the respond() method, rather than invoking the function itself. Appropriate success and error messages can be configured with the createSuccessResponse(invocationId,message) and createErrorResponse(invocationId,functionsErrorType,errorMsg) methods.

See FunctionInvokeMock Example Implementation.

MockFunctionInvocationFactory Methods

The following are methods for MockFunctionInvocationFactory.

createErrorResponse(invocationId, functionsErrorType, errorMsg)
Generate a response for an error condition during mock testing of Salesforce Functions.
createSuccessResponse(invocationId, response)
Generate a response for a successful mock test of Salesforce Functions.

createErrorResponse(invocationId, functionsErrorType, errMsg)
Generate a response for an error condition during mock testing of Salesforce Functions.

Signature
public static functions.FunctionInvocation createErrorResponse(String invocationId, functions.FunctionErrorType functionsErrorType, String errMsg)

Parameters
invocationId
Type: String
The ID associated with a call to either the synchronous or asynchronous Function.invoke() method.

functionsErrorType
Type: FunctionErrorType Enum
The error type of FunctionInvocationError.

eerrMsg
Type: String
The error message.

Return Value
Type: FunctionInvocation Interface

createSuccessResponse(invocationId, response)
Generate a response for a successful mock test of Salesforce Functions.

Signature
public static functions.FunctionInvocation createSuccessResponse(String invocationId, String response)

Parameters
invocationId
Type: String
The ID associated with a call to either the synchronous or asynchronous Function.invoke() method.

response
Type: String
The message indicating success.
Return Value
Type: FunctionInvocation Interface

Invocable Namespace

The Invocable namespace provides classes for calling invocable actions from Apex. These classes are in the Invocable namespace.

IN THIS SECTION:
  Action Class
  Contains methods to create, update, and retrieve information about invocable actions.
  Action.Error Class
  Contains methods to retrieve errors returned by invocable actions.
  Action.Result Class
  Contains methods to retrieve results from invocable actions called from Apex code.

Action Class
Contains methods to create, update, and retrieve information about invocable actions.

Namespace
Invocable

IN THIS SECTION:
  Action Methods

SEE ALSO:
  Apex Developer Guide: InvocableMethod Annotation

Action Methods
These methods are for Action.

IN THIS SECTION:
  addInvocation()
  Creates an empty invocation in preparation for calling an invocable action. After you create the invocation, you can add parameters to the invocation.
  clearInvocations()
  Clears the existing invocations from the action.
  clone()
  Creates a copy of the Invocable.Action.
createCustomAction(type, namespace, name)
Creates a wrapper for a custom invocable action in a specified namespace.

createCustomAction(type, name)
Creates a wrapper for a custom invocable action.

createStandardAction(type)
Creates a wrapper for a standard invocable action.

getName()
Gets the name of an invocable action.

getNamespace()
Gets the namespace of a custom invocable action.

getType()
Gets the type of an invocable action.

invoke()
Invokes an invocable action from Apex code.

isStandard()
Determines whether an invocable action is a standard invocable action.

setInvocationParameter(parameterName, parameterValue)
Sets a value for an invocable action parameter.

setInvocations(invocations)
Initializes the invocations for an action from a pre-existing list of invocations.

addInvocation()
Creates an empty invocation in preparation for calling an invocable action. After you create the invocation, you can add parameters to the invocation.

Signature
public Invocable.Action addInvocation()

Return Value
Type: Invocable.Action on page 2232

clearInvocations()
Clears the existing invocations from the action.

Signature
public Invocable.Action clearInvocations()

Return Value
Type: Invocable.Action on page 2232
**clone()**

Creates a copy of the Invocable.Action.

**Signature**

```java
public Object clone()
```

**Return Value**

Type: Object

---

**createCustomAction(type, namespace, name)**

Creates a wrapper for a custom invocable action in a specified namespace.

**Signature**

```java
public static Invocable.Action createCustomAction(String type, String namespace, String name)
```

**Parameters**

- **type**
  - Type: String
  - Type of invocable action.
- **namespace**
  - Type: String
  - Namespace where the invocable action is located.
- **name**
  - Type: String
  - Name for the custom invocable action.

**Return Value**

Type: Invocable.Action

---

**createCustomAction(type, name)**

Creates a wrapper for a custom invocable action.

**Signature**

```java
public static Invocable.Action createCustomAction(String type, String name)
```

**Parameters**

- **type**
  - Type: String
Type of invocable action.

name
  Type: String
  Name for the custom invocable action.

Return Value
Type: Invocable.Action

createStandardAction(type)
Creates a wrapper for a standard invocable action.

Signature
public static Invocable.Action createStandardAction(String type)

Parameters
type
  Type: String
  Type of invocable action.

Return Value
Type: Invocable.Action

getName()
Gets the name of an invocable action.

Signature
public String getName()

Return Value
Type: String
Name of the invocable action.

getNamespace()
Gets the namespace of a custom invocable action.

Signature
public String getNamespace()
Return Value
Type: String
Namespace of the custom invocable action.

**getType()**
Gets the type of an invocable action.

**Signature**
```
public String getType()
```

Return Value
Type: String
Type of invocable action.

**invoke()**
Invokes an invocable action from Apex code.

**Signature**
```
public List<Invocable.Action.Result> invoke()
```

Return Value
Type: List<Invocable.Action.Result>

**isStandard()**
Determines whether an invocable action is a standard invocable action.

**Signature**
```
public Boolean isStandard()
```

Return Value
Type: Boolean
This method returns `true` if the invocable action is a standard invocable action.

**setInvocationParameter(parameterName, parameterValue)**
Sets a value for an invocable action parameter.
**Signature**

```java
public Invocable.Action setInvocationParameter(String parameterName, Object parameterValue)
```

**Parameters**

- `parameterName`
  - Type: `String`
  - Name of the invocable action parameter to set.

- `parameterValue`
  - Type: `Object`
  - Value to set the invocable action parameter to.

**Return Value**

Type: `Invocable.Action` on page 2232

**setInvocations(invocations)**

Initializes the invocations for an action from a pre-existing list of invocations.

**Signature**

```java
public Invocable.Action setInvocations(List<Map<String,ANY>> invocations)
```

**Parameters**

- `invocations`
  - Type: `List` on page 3177 `<Map` on page 3198 `<String` on page 3402,ANY`>`
  - List of invocations for the invocable action.

**Return Value**

Type: `Invocable.Action` on page 2232

**Action.Error Class**

Contains methods to retrieve errors returned by invocable actions.

**Namespace**

`Invocable`

IN THIS SECTION:

- Action.Error Methods
Action.Error Methods

These methods are for Action.Error.

IN THIS SECTION:

- **getCode()**: Gets the error code returned by an invocable action.
- **getMessage()**: Gets the error message returned by an invocable action.

**clone()**


**Signature**

```java
public Object clone()
```

**Return Value**

Type: Object

**getCode()**

Gets the error code returned by an invocable action.

**Signature**

```java
public String getCode()
```

**Return Value**

Type: String

**getMessage()**

Gets the error message returned by an invocable action.

**Signature**

```java
public String getMessage()
```

**Return Value**

Type: String
**Action.Result Class**

Contains methods to retrieve results from invocable actions called from Apex code.

**Namespace**

Invocable

**IN THIS SECTION:**

*Action.Result Methods*

**Action.Result Methods**

The methods are for `Action.Result`.

**IN THIS SECTION:**

- `clone()`
- `getAction()`
  Gets the invocable action that was invoked and caused a result to be returned.
- `getErrors()`
  Gets a list of errors that were returned by an invocable action.
- `getInvocationParameters()`
  Gets a list of the parameter values set for an invocable action. This method returns a list that contains the input parameter values for each invocation of an action. Each map in the list contains a key for the name of each input parameter.
- `getOutputParameters()`
  Gets a list of the parameter values returned by an invocable action. This method returns a list that contains the result for each invocation of an action. Each map in the list contains a key for the name of each output parameter.
- `isSuccess()`
  Determines if an invocable action ran without errors.

**clone()**


**Signature**

```java
public Object clone()
```

**Return Value**

Type: Object
**getAction()**
Gets the invocable action that was invoked and caused a result to be returned.

**Signature**

```java
public Invocable.Action getAction()
```

**Return Value**
Type: `Invocable.Action` on page 2232

**getErrors()**
Gets a list of errors that were returned by an invocable action.

**Signature**

```java
public List<Invocable.Action.Error> getErrors()
```

**Return Value**
Type: `List<Invocable.Action.Error>` on page 2237

**getInvocationParameters()**
Gets a list of the parameter values set for an invocable action. This method returns a list that contains the input parameter values for each invocation of an action. Each map in the list contains a key for the name of each input parameter.

**Signature**

```java
public Map<String, Object> getInvocationParameters()
```

**Return Value**
Type: `Map<String, Object>` on page 3198

**getOutputParameters()**
Gets a list of the parameter values returned by an invocable action. This method returns a list that contains the result for each invocation of an action. Each map in the list contains a key for the name of each output parameter.

**Signature**

```java
public Map<String, Object> getOutputParameters()
```

**Return Value**
Type: `Map<String, Object>` on page 3198
**isSuccess()**
Determines if an invocable action ran without errors.

**Signature**

```java
public Boolean isSuccess()
```

**Return Value**

Type: Boolean

This method returns `true` if the invocable action ran successfully.

---

**KbManagement Namespace**

The `KbManagement` namespace provides a class for managing knowledge articles.

The following is the class in the `KbManagement` namespace.

---

**IN THIS SECTION:**

- PublishingService Class
  - Use the methods in the `KbManagement.PublishingService` class to manage the lifecycle of an article and its translations.

**PublishingService Class**

Use the methods in the `KbManagement.PublishingService` class to manage the lifecycle of an article and its translations.

**Namespace**

`KbManagement`

**Usage**

Use the methods in the `KbManagement.PublishingService` class to manage the following parts of the lifecycle of an article and its translations:

- Publishing
- Updating
- Retrieving
- Deleting
- Submitting for translation
- Setting a translation to complete or incomplete status
- Archiving
- Assigning review tasks for draft articles or translations

**Note:** Date values are based on GMT.
To use the methods in this class, you must enable Salesforce Knowledge. See Salesforce Knowledge Implementation Guide for more information on setting up Salesforce Knowledge.

**PublishingService Methods**

The following are methods for `PublishingService`. All methods are static.

**IN THIS SECTION:**

- `archiveOnlineArticle(articleId, scheduledDate)`: Archives an online version of an article. If the specified scheduledDate is null, the article is archived immediately. Otherwise, it archives the article on the scheduled date.
- `assignDraftArticleTask(articleId, assigneeId, instructions, dueDate, sendEmailNotification)`: Assigns a review task related to a draft article.
- `assignDraftTranslationTask(articleVersionId, assigneeId, instructions, dueDate, sendEmailNotification)`: Assigns a review task related to a draft translation.
- `cancelScheduledArchivingOfArticle(articleId)`: Cancels the scheduled archiving of an online article.
- `cancelScheduledPublicationOfArticle(articleId)`: Cancels the scheduled publication of a draft article.
- `completeTranslation(articleVersionId)`: Puts a translation in a completed state that is ready to publish.
- `deleteArchivedArticle(articleId)`: Deletes an archived article.
- `deleteArchivedArticleVersion(articleId, versionNumber)`: Deletes a specific archived version of a published article.
- `deleteDraftArticle(articleId)`: Deletes a draft article.
- `deleteDraftTranslation(articleVersionId)`: Deletes a draft translation.
- `editArchivedArticle(articleId)`: Creates a draft article from the archived primary version and returns the new draft primary version ID of the article.
- `editOnlineArticle(articleId, unpublish)`: Creates a draft article from the online version and returns the new draft primary version ID of the article. Also, unpublishes the online article, if `unpublish` is set to `true`.
- `editPublishedTranslation(articleId, language, unpublish)`: Creates a draft version of the online translation for a specific language and returns the new draft primary version ID of the article. Also, unpublishes the article, if set to `true`.
- `publishArticle(articleId, flagAsNew)`: Publishes an article. If `flagAsNew` is set to `true`, the article is published as a major version.
- `restoreOldVersion(articleId, versionNumber)`: Creates a draft article from an existing online article based on the specified archived version of the article and returns the article version ID.
scheduleForPublication(articleId, scheduledDate)
Schedules the article for publication as a major version. If the specified date is null, the article is published immediately.

setTranslationToIncomplete(articleVersionId)
Sets a draft translation that is ready for publication back to “in progress” status.

submitForTranslation(articleId, language, assigneeId, dueDate)
Submits an article for translation to the specified language. Also assigns the specified user and due date to the submittal and returns new ID of the draft translation.

archiveOnlineArticle(articleId, scheduledDate)
Archives an online version of an article. If the specified scheduledDate is null, the article is archived immediately. Otherwise, it archives the article on the scheduled date.

Signature
public static Void archiveOnlineArticle(String articleId, Datetime scheduledDate)

Parameters

articleId
Type: String

scheduledDate
Type: Datetime

Return Value
Type: Void

Example
String articleId = 'Insert article ID';
Datetime scheduledDate = Datetime.newInstanceGmt(2012, 12, 13, 13, 30, 0);
KbManagement.PublishingService.archiveOnlineArticle(articleId, scheduledDate);

assignDraftArticleTask(articleId, assigneeId, instructions, dueDate, sendEmailNotification)
Assigns a review task related to a draft article.

Signature
public static Void assignDraftArticleTask(String articleId, String assigneeId, String instructions, Datetime dueDate, Boolean sendEmailNotification)

Parameters

articleId
Type: String
assigneeId
Type: String

instructions
Type: String

dueDate
Type: Datetime

sendEmailNotification
Type: Boolean

Return Value
Type: Void

Example

String articleId = 'Insert article ID';
String assigneeId = '';
String instructions = 'Please review this draft.';
Datetime dueDate = Datetime.newInstanceGmt(2012, 12, 1);
KbManagement.PublishingService.assignDraftArticleTask(articleId, assigneeId, instructions, dueDate, true);

assignDraftTranslationTask(articleVersionId, assigneeId, instructions, dueDate, sendEmailNotification)

Assigns a review task related to a draft translation.

Signature

public static Void assignDraftTranslationTask(String articleVersionId, String assigneeId, String instructions, Datetime dueDate, Boolean sendEmailNotification)

Parameters

articleVersionId
Type: String

assigneeId
Type: String

instructions
Type: String

dueDate
Type: Datetime

sendEmailNotification
Type: Boolean
Return Value
Type: Void

Example
String articleId = 'Insert article ID';
String assigneeId = 'Insert assignee ID';
String instructions = 'Please review this draft.';
Datetime dueDate = Datetime.newInstanceGmt(2012, 12, 1);
KbManagement.PublishingService.assignDraftTranslationTask(articleId, assigneeId, instructions, dueDate, true);

cancelScheduledArchivingOfArticle(articleId)
Cancels the scheduled archiving of an online article.

Signature
public static Void cancelScheduledArchivingOfArticle(String articleId)

Parameters
articleId
Type: String

Return Value
Type: Void

Example
String articleId = 'Insert article ID';
KbManagement.PublishingService.cancelScheduledArchivingOfArticle(articleId);

cancelScheduledPublicationOfArticle(articleId)
Cancels the scheduled publication of a draft article.

Signature
public static Void cancelScheduledPublicationOfArticle(String articleId)

Parameters
articleId
Type: String

Return Value
Type: Void
Example

```java
String articleId = 'Insert article ID';
KbManagement.PublishingService.cancelScheduledPublicationOfArticle (articleId);
```

cancelScheduledPublicationOfArticle(articleId)

Puts a translation in a completed state that is ready to publish.

Signature

```java
public static Void cancelScheduledPublicationOfArticle(String articleId)
```

Parameters

- **articleId**
  - Type: String

Return Value

Type: Void

Example

```java
String articleId = 'Insert article ID';
KbManagement.PublishingService.cancelScheduledPublicationOfArticle (articleId);
```

completeTranslation(articleVersionId)

Puts a translation in a completed state that is ready to publish.

Signature

```java
public static Void completeTranslation(String articleVersionId)
```

Parameters

- **articleVersionId**
  - Type: String

Return Value

Type: Void

Example

```java
String articleVersionId = 'Insert article ID';
KbManagement.PublishingService.completeTranslation(articleVersionId);
```

deleteArchivedArticle(articleId)

Deletes an archived article.

Signature

```java
public static Void deleteArchivedArticle(String articleId)
```

Parameters

- **articleId**
  - Type: String

Return Value

Type: Void

Example

```java
String articleId = 'Insert article ID';
KbManagement.PublishingService.deleteArchivedArticle(articleId);
```
**deleteArchivedArticleVersion(articleId, versionNumber)**

Deletes a specific archived version of a published article.

**Signature**

```java
public static Void deleteArchivedArticleVersion(String articleId, Integer versionNumber)
```

**Parameters**

- **articleId**
  Type: String
- **versionNumber**
  Type: Integer

**Return Value**

Type: Void

**Example**

```java
String articleId = 'Insert article ID';
Integer versionNumber = 1;
KbManagement.PublishingService.deleteArchivedArticleVersion(articleId, versionNumber);
```

**deleteDraftArticle(articleId)**

Deletes a draft article.

**Signature**

```java
public static Void deleteDraftArticle(String articleId)
```

**Parameters**

- **articleId**
  Type: String

**Return Value**

Type: Void

**Example**

```java
String articleId = 'Insert article ID';
KbManagement.PublishingService.deleteDraftArticle(articleId);
```

**deleteDraftTranslation(articleVersionId)**

Deletes a draft translation.

**Example**

```java
String articleId = 'Insert article ID';
KbManagement.PublishingService.deleteDraftTranslation(articleVersionId);
```
Signature

public static Void deleteDraftTranslation(String articleVersionId)

Parameters

articleVersionId
    Type: String

Return Value

Type: Void

Example

String articleVersionId = 'Insert article ID';
KbManagement.PublishingService.deleteDraftTranslation (articleVersionId);

editArchivedArticle(articleId)

Creates a draft article from the archived primary version and returns the new draft primary version ID of the article.

Signature

public static String editArchivedArticle(String articleId)

Parameters

articleId
    Type: String

Return Value

Type: String

Example

String articleId = 'Insert article ID';
String id = KbManagement.PublishingService.editArchivedArticle(articleId);

editOnlineArticle(articleId, unpublish)

Creates a draft article from the online version and returns the new draft primary version ID of the article. Also, unpublishes the online article, if unpublish is set to true.

Signature

public static String editOnlineArticle(String articleId, Boolean unpublish)
Parameters

- **articleId**
  - Type: String
- **unpublish**
  - Type: Boolean

Return Value

- Type: String

Example

```java
String articleId = 'Insert article ID';
String id = KbManagement.PublishingService.editOnlineArticle(articleId, true);
```

editPublishedTranslation(articleId, language, unpublish)

Creates a draft version of the online translation for a specific language and returns the new draft primary version ID of the article. Also, unpublishes the article, if set to `true`.

Signature

```java
public static String editPublishedTranslation(String articleId, String language, Boolean unpublish)
```

Parameters

- **articleId**
  - Type: String
- **language**
  - Type: String
- **unpublish**
  - Type: Boolean

Return Value

- Type: String

Example

```java
String articleId = 'Insert article ID';
String language = 'fr';
String id = KbManagement.PublishingService.editPublishedTranslation(articleId, language, true);
```

publishArticle(articleId, flagAsNew)

Publishes an article. If `flagAsNew` is set to `true`, the article is published as a major version.
Signature

`public static Void publishArticle(String articleId, Boolean flagAsNew)`

Parameters

- `articleId`
  - Type: `String`
- `flagAsNew`
  - Type: `Boolean`

Return Value

Type: `Void`

Example

```java
String articleId = 'Insert article ID';
KbManagement.PublishingService.publishArticle(articleId, true);
```

`restoreOldVersion(articleId, versionNumber)`

Creates a draft article from an existing online article based on the specified archived version of the article and returns the article version ID.

Signature

`public static String restoreOldVersion(String articleId, Integer versionNumber)`

Parameters

- `articleId`
  - Type: `String`
- `versionNumber`
  - Type: `Integer`

Return Value

Type: `String`

Example

```java
String articleId = 'Insert article ID';
String id = KbManagement.PublishingService.restoreOldVersion (articleId, 1);
```

`scheduleForPublication(articleId, scheduledDate)`

Schedules the article for publication as a major version. If the specified date is null, the article is published immediately.
**Signature**

public static Void scheduleForPublication(String articleId, Datetime scheduledDate)

**Parameters**

articleId  
Type: String  
scheduledDate  
Type: Datetime

**Return Value**

Type: Void

**Example**

```java
String articleId = 'Insert article ID';
Datetime scheduledDate = Datetime.newInstanceGmt(2012, 12, 1, 13, 30, 0);
KbManagement.PublishingService.scheduleForPublication(articleId, scheduledDate);
```

setTranslationToIncomplete(articleVersionId)

Sets a draft translation that is ready for publication back to "in progress" status.

**Signature**

public static Void setTranslationToIncomplete(String articleVersionId)

**Parameters**

articleVersionId  
Type: String

**Return Value**

Type: Void

**Example**

```java
String articleVersionId = 'Insert article ID';
KbManagement.PublishingService.setTranslationToIncomplete(articleVersionId);
```

submitForTranslation(articleId, language, assigneeId, dueDate)

Submits an article for translation to the specified language. Also assigns the specified user and due date to the submittal and returns new ID of the draft translation.
Signature

public static String submitForTranslation(String articleId, String language, String assigneeId, Datetime dueDate)

Parameters

articleId
Type: String

language
Type: String

assigneeId
Type: String

dueDate
Type: Datetime

Return Value

Type: String

Example

String articleId = 'Insert article ID';
String language = 'fr';
String assigneeId = 'Insert assignee ID';
Datetime dueDate = Datetime.newInstanceGmt(2012, 12,1);
String id = KbManagement.PublishingService.submitForTranslation(articleId, language, assigneeId, dueDate);

LxScheduler Namespace

The LxScheduler namespace provides an interface and classes for integrating Salesforce Scheduler with external calendars. The following are the classes and the interface in the LxScheduler namespace.

IN THIS SECTION:

GetAppointmentCandidatesInput Class
Contains information about the available service resources (appointment candidates) based on work type group and service territories.

GetAppointmentCandidatesInputBuilder Class
Contains methods to build an instance of the lxscheduler.GetAppointmentCandidatesInput class.

GetAppointmentSlotsInput Class
Contains information about the available appointment time slots for a resource based on given work type group and territories.

GetAppointmentSlotsInputBuilder Class
Contains methods to build an instance of the lxscheduler.GetAppointmentSlotsInput class.

SchedulerResources Class
Contains methods that holds the business logic to get resources availability.
SkillRequirement Class
Contains information about the set of skills that are required to complete a particular task for a work type.

SkillRequirementBuilder Class
Contains methods to build an instance of the `lxscheduler.SkillRequirement` class.

WorkType Class
Contains information about the type of work to be performed.

WorkTypeBuilder Class
Contains methods to build an instance of the `lxscheduler.WorkType` class.

ServiceResourceScheduleHandler Interface
Allows an implementing class to check external calendar events to find already booked time slots for the requested service resources. This interface is part of Salesforce Scheduler.

ServiceAppointmentRequestInfo Class
Represents the list of parameters that are passed to the `ServiceResourceScheduleHandler` interface. This class is implemented internally by Apex.

ServiceResourceInfo Class
Contains information about a service resource.

ServiceResourceSchedule Class
Use this class to pass results from your implemented Apex class to the `ServiceResourceScheduleHandler` interface methods.

UnavailableTimeslot Class
Use this class to pass the unavailable time slots to the `lxscheduler.ServiceResourceSchedule` class.

SEE ALSO:

Apex Interface Implementation Limitations and Error Codes

---

GetAppointmentCandidatesInput Class

Contains information about the available service resources (appointment candidates) based on work type group and service territories. Set up Salesforce Scheduler before making requests. This setup includes creating or configuring Service Resources, Service Territory Members, Work Type Groups, Work Types, Work Type Group Members, and Service Territory Work Types. See Set Up Salesforce Scheduler for more information.

The appointment time slots are determined based on multiple factors, such as field values, scheduled appointments, absences, Scheduler Settings, and Scheduling Policies to determine available time slots. See How Salesforce Scheduler Determines Available Time Slots for more information.

The following factors are considered for returning start time and end time of resources.

**Resource Availability**
Determined using service territory member, service territory, work type, and account operating hours fields.

**Resource Unavailability**
Determined by resource absences, existing appointments that the resource is assigned to. The resource must be marked as a required resource for the appointment with a status that isn’t in closed, canceled, or completed.

**Appointment Start Time Interval in the Scheduling Policy**
Appointment start time interval field in the Scheduling Policy is used to determine when the appointment can start. This interval can be 5, 10, 15, 20, 30, or 60. By default, it’s set to 15.
Work Type Duration

The end time is calculated as start time + duration of the work type.

Note: If asset scheduling is enabled, the response also includes asset-based candidates.

Namespace

LxScheduler

Usage

The constructor for this class can’t be called directly. Create an instance of this class using the GetAppointmentCandidatesInputBuilder.build() method.

This example shows how to get a list of available appointment candidates based on workTypeGroupId:

```java
//Build input for GetAppointmentCandidates API
lxscheduler.GetAppointmentCandidatesInput input = new lxscheduler.GetAppointmentCandidatesInputBuilder()
    .setWorkTypeGroupId('0VSRM0000000ABc4AM')
    .setTerritoryIds(new List<String>{'0HhRM0000000FXd0AM'})
    .setStartTime(System.now().format('yyyy-MM-ddERSIONal: T\'HH:mm:ssZ','America/New_York'))
    .setEndTime(System.now().addDays(5).format('yyyy-MM-dd\'T\'HH:mm:ssZ','America/New_York'))
    .setAccountId('001RM0000053iQgYAI')
    .setSchedulingPolicyId('0VrRM00000000Bx')
    .setApiVersion(Double.valueOf('50.0'))
    .build();

String response = lxscheduler.SchedulerResources.getAppointmentCandidates(input);
```

This example shows how to get a list of available appointment candidates based on workType:

```java
//Build WorkType
lxscheduler.WorkType workType = new lxscheduler.WorkTypeBuilder()
    .setId('08qRM0000000G9RYAU')
    .build();

lxscheduler.GetAppointmentCandidatesInput input = new lxscheduler.GetAppointmentCandidatesInputBuilder()
    .setWorkType(workType)
    .setTerritoryIds(new List<String>{'0HhRM0000000FXd0AM'})
    .setStartTime(System.now().format('yyyy-MM-dd\'T\'HH:mm:ssZ','America/New_York'))
    .setEndTime(System.now().addDays(5).format('yyyy-MM-dd\'T\'HH:mm:ssZ','America/New_York'))
    .setAccountId('001RM0000053iQgYAI')
    .setSchedulingPolicyId('0VrRM00000000Bx')
    .setApiVersion(Double.valueOf('50.0'))
    .build();

String response = lxscheduler.SchedulerResources.getAppointmentCandidates(input);
```
This example shows how to get a list of available candidate appointments based on `durationInMinutes` and without the `workTypeGroupId` or `workType` fields:

```java
//Build SkillRequirement
lxscheduler.SkillRequirement skillReq = new lxscheduler.SkillRequirementBuilder()
    .setSkillId('0C5RM0000004EZS0A2')
    .setSkillLevel(90)
    .build();

//Build WorkType
lxscheduler.WorkType workType = new lxscheduler.WorkTypeBuilder()
    .setDurationInMinutes(15)
    .setBlockTimeBeforeAppointmentInMinutes(5)
    .setBlockTimeAfterAppointmentInMinutes(5)
    .setTimeFrameStartInMinutes(10080)
    .setTimeFrameEndInMinutes(40320)
    .setOperatingHoursId('0OHRM0000000FmG4AU')
    .setSkillRequirements(new List<lxscheduler.SkillRequirement>{skillReq})
    .build();

lxscheduler.GetAppointmentCandidatesInput input = new
lxscheduler.GetAppointmentCandidatesInputBuilder()
    .setWorkType(workType)
    .setTerritoryIds(new List<String>{'0HhRM0000000FXd0AM'})
    .setSchedulingPolicyId('0VrRM00000000Bx')
    .setApiVersion(Double.valueOf('50.0'))
    .build();

String response = lxscheduler.SchedulerResources.getAppointmentCandidates(input);
```

This example shows a sample response of a list of available candidates:

```json
[
  {
    "startTime": "2021-02-16T16:15:00.000+0000",
    "endTime": "2021-02-16T16:16:00.000+0000",
    "resources": ["0Hnxx00000004C9BCAU"],
    "territoryId": "0Hhxx00000004C92CAE"
  },
  {
    "startTime": "2021-02-16T16:30:00.000+0000",
    "endTime": "2021-02-16T16:31:00.000+0000",
    "resources": ["0Hnxx00000004C9BCAU"],
    "territoryId": "0Hhxx00000004C92CAE"
  }
]
```
GetAppointmentCandidatesInputBuilder Class

Contains methods to build an instance of the lxscheduler.GetAppointmentCandidatesInput class.

A Builder object is obtained by invoking one of the GetAppointmentCandidatesInputBuilder methods defined by the GetAppointmentCandidatesInput class.

Namespace

LxScheduler

IN THIS SECTION:

GetAppointmentCandidatesInputBuilder Methods

GetAppointmentCandidatesInputBuilder Methods

The following are methods for GetAppointmentCandidatesInputBuilder.

IN THIS SECTION:

- `setAccountId(accountId)`: Sets the ID of the associated account for which you want to create the appointments.
- `setAllowConcurrent(allowConcurrent)`: Allows the scheduling of concurrent appointments.
- `setApiVersion(apiVersion)`: Sets the API version of the business logic for the getAppointmentCandidates method.
- `setCorrelationId(correlationId)`: Sets the correlation ID.
- `setEndTime(endTime)`: Sets the scheduling end time.
- `setEngagementChannelTypeIds(engagementChannelTypeIds)`: Sets an engagement channel type.
- `setFilterByResources(filterByResources)`: Enables filtering resources using a comma-separated list of service resource IDs.
- `setResourceLimitApptDistribution(resourceLimitApptDistribution)`: Sets the number of service resources to show during appointment scheduling.
- `setSchedulingPolicyId(schedulingPolicyId)`: Sets the ID of the AppointmentSchedulingPolicy object.
- `setStartTime(startTime)`: Sets the scheduling start time to the specified time.
- `setTerritoryIds(territoryIds)`: Sets the service territory IDs.
setWorkType(workType)
Sets the type of work to be performed.

setWorkTypeGroupId(workTypeGroupId)
Sets the ID of the work type group.

**build()**
Returns an instance of the `lxscheduler.GetAppointmentCandidatesInput` object.

**Signature**
```java
public lxscheduler.GetAppointmentCandidatesInput build()
```

**Return Value**
Type: `lxscheduler.GetAppointmentCandidatesInput`

**setAccountId(accountId)**
Sets the ID of the associated account for which you want to create the appointments.

**Signature**
```java
public lxscheduler.GetAppointmentCandidatesInputBuilder setAccountId(String accountId)
```

**Parameters**
- `accountId`
  Type: `String`

**Return Value**
Type: `LxScheduler.GetAppointmentCandidatesInputBuilder`

**setAllowConcurrent(allowConcurrent)**
Allows the scheduling of concurrent appointments.

**Signature**
```java
public lxscheduler.GetAppointmentCandidatesInputBuilder setAllowConcurrent(Boolean allowConcurrent)
```

**Parameters**
- `allowConcurrent`
  Type: `Boolean`
  If true, allows scheduling of concurrent appointments in a time slot. The default is false.
  Available in API version 47.0 and later.
Return Value
Type: LxScheduler.GetAppointmentCandidatesInputBuilder

**setApiVersion(apiVersion)**
Sets the API version of the business logic for the `getAppointmentCandidates` method.

**Signature**
`public lxscheduler.GetAppointmentCandidatesInputBuilder setApiVersion(Double apiVersion)`

**Parameters**
- `apiVersion`
  Type: `Double`

**Usage**
The specified parameter must use the correct API version. For example, if API version is set to 45.0 and `filterByResources` is set (which is available in API version 51.0 and later), then this field is ignored. If no API version or incorrect API version is passed in the request body, by default the latest version is used.

⚠️ **Note:** The API is available since version 45.0.

Return Value
Type: LxScheduler.GetAppointmentCandidatesInputBuilder

**setCorrelationId(correlationId)**
Sets the correlation ID.

**Signature**
`public lxscheduler.GetAppointmentCandidatesInputBuilder setCorrelationId(String correlationId)`

**Parameters**
- `correlationId`
  Type: `String`
  ID to pass custom information to the `ServiceResourceScheduleHandler` Apex interface. For example, you can use the correlation ID to identify the app, website, or any other external system that calls this Apex interface implementation. If you don’t pass a custom value, a randomly generated identifier is passed. Available in API version 53.0 and later.

Return Value
Type: LxScheduler.GetAppointmentCandidatesInputBuilder
**setEndTime(endTime)**
Sets the scheduling end time.

**Signature**

```java
public lxscheduler.GetAppointmentCandidatesInputBuilder setEndTime(String endTime)
```

**Parameters**

`endTime`
Type: `String`
The latest time that a time slot can end (inclusive).

⚠️ **Note:** If end time is not specified, it defaults to 31 days.

**Usage**
The specified string should use the standard date format 
"yyyy-MM-dd'T'HH:mm:ssZ" in the local time zone. Defaults to the user’s time zone.

**Return Value**
Type: `LxScheduler.GetAppointmentCandidatesInputBuilder`

**setEngagementChannelTypeIds(engagementChannelTypeIds)**
Sets an engagement channel type.

**Signature**

```java
public lxscheduler.GetAppointmentCandidatesInputBuilder setEngagementChannelTypeIds(List<String> engagementChannelTypeIds)
```

**Parameters**

`engagementChannelTypeIds`
Type: `List<String>`
The ID of the engagement channel type record. The availability of service resources is filtered based on the engagement channel type selected. This field is available in API version 56.0 and later.

⚠️ **Note:** This field supports only one engagement channel type ID.

**Return Value**
Type: `LxScheduler.GetAppointmentCandidatesInputBuilder`

**Usage**
You can use engagement channel types only in these cases:
The **Schedule Appointments Using Engagement Channels** setting is enabled in Salesforce Scheduler Settings in your Salesforce org.

Shifts are defined in the scheduling policy. For more information on setting up shifts in scheduling policy, see Define Shift Rules in Scheduling Policy.

**Note:** Engagement channel types are not supported with operating-hours rules in the scheduling policy.

### setFilterByResources(filterByResources)

Enables filtering resources using a comma-separated list of service resource IDs.

**Signature**

```java
public LxScheduler.GetAppointmentCandidatesInputBuilder setFilterByResources(List<String> filterByResources)
```

**Parameters**

- **filterByResources**
  - Type: `List<String>`
  - Gets only eligible resources that are both in the list and in the selected service territory sorted by the order in which the resource IDs are passed. This field is available in API version 51.0 and later.

**Return Value**

Type: `LxScheduler.GetAppointmentCandidatesInputBuilder`

### setResourceLimitApptDistribution(resourceLimitApptDistribution)

Sets the number of service resources to show during appointment scheduling.

**Signature**

```java
```

**Parameters**

- **resourceLimitApptDistribution**
  - Type: `Integer`
  - Specify the maximum number of service resources that you want to show during appointment scheduling when appointment distribution is enabled. Available in API version 53.0 and later.

**Return Value**

Type: `LxScheduler.GetAppointmentCandidatesInputBuilder`
**setSchedulingPolicyId(schedulingPolicyId)**
Sets the ID of the AppointmentSchedulingPolicy object.

**Signature**

```java
public lxscheduler.GetAppointmentCandidatesInputBuilder setSchedulingPolicyId(String schedulingPolicyId)
```

**Parameters**

- **schedulingPolicyId**
  - **Type:** String
  - The ID of the AppointmentSchedulingPolicy object. If no scheduling policy is passed in the request body, the default configurations are used.

**Return Value**

- **Type:** LxScheduler.GetAppointmentCandidatesInputBuilder

**setStartTime(startTime)**
Sets the scheduling start time to the specified time.

**Signature**

```java
public lxscheduler.GetAppointmentCandidatesInputBuilder setStartTime(String startTime)
```

**Parameters**

- **startTime**
  - **Type:** String
  - The earliest time that a time slot can begin (inclusive). You can also use a time from the past.

**Usage**

The specified string should use the standard date format "'yyyy-MM-dd'T'HH:mm:ssZ'" in the local time zone. Defaults to the user’s time zone.

**Return Value**

- **Type:** LxScheduler.GetAppointmentCandidatesInputBuilder

**setTerritoryIds(territoryIds)**
Sets the service territory IDs.
Signature

```java
public lxscheduler.GetAppointmentCandidatesInputBuilder setTerritoryIds(List<String> territoryIds)
```

Parameters

territoryIds
Type: List<String>
List of service territory IDs, where the work that is being requested is performed. This is a required field.

Return Value
Type: LxScheduler.GetAppointmentCandidatesInputBuilder

**setWorkType(workType)**
Sets the type of work to be performed.

Signature

```java
public lxscheduler.GetAppointmentCandidatesInputBuilder setWorkType(lxscheduler.WorkType workType)
```

Parameters

workType
Type: lxscheduler.WorkType
This method takes input as an instance of the lxscheduler.WorkType class. Build the instance of the input class using the lxscheduler.WorkTypeBuilder class.

Required if `workTypeGroupId` is not given. If id of the `workType` is given, the rest of `workType` fields are optional.

Usage

Return Value
Type: LxScheduler.GetAppointmentCandidatesInputBuilder

**setWorkTypeGroupId(workTypeGroupId)**
Sets the ID of the work type group.

Signature

```java
public lxscheduler.GetAppointmentCandidatesInputBuilder setWorkTypeGroupId(String workTypeGroupId)
```
Parameters

workTypeGroupId
Type: String
The ID of the work type group containing the work types that are being performed. Required if workType is not given. If workType is given, then you must provide either id or durationInMinutes, but not both.

Return Value
Type: LxScheduler.GetAppointmentCandidatesInputBuilder

GetAppointmentSlotsInput Class
Contains information about the available appointment time slots for a resource based on given work type group and territories.

The appointment time slots are determined based on your Salesforce Scheduler data model configurations. Here are some prerequisites that you can consider while setting up data.

• Set up Salesforce Scheduler before making your requests. The setup includes creating or configuring Service Resources, Service Territory Members, Work Type Groups, Work Types, Work Type Group Members, and Service Territory Work Types. See Manage Business Information in Salesforce Scheduler for more information.

• Configure a work type mapped for each territory in the request body via Service Territory Work Type. Map the same work type to the work type group, via work type group member.

The following factors affect how time slots are calculated and returned.

• Timezones that differ across operating hours are handled and results are always returned in UTC.
• The resource must be marked as a required resource on the assigned resource object.
• The resource is considered unavailable if the status categories of the resource assigned to service appointments are other than Canceled, Cannot Complete, and Completed.
• Resource Absences of all types are considered unavailable from start to end.
• The following fields of Work Type records, if configured, are used to fine-tune time slot requirements. For more information, see Create Work Types in Salesforce Scheduler.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Timeframe Start</td>
<td>Time slots sooner than current time + Timeframe Start aren't returned.</td>
</tr>
<tr>
<td>Timeframe End</td>
<td>Time slots later than current time + Timeframe End aren't returned.</td>
</tr>
<tr>
<td>Block Time Before Appointment</td>
<td>The time period before the appointment is considered as unavailable.</td>
</tr>
<tr>
<td>Block Time After Appointment</td>
<td>The time period after the appointment is considered as unavailable.</td>
</tr>
<tr>
<td>Operating Hours</td>
<td>The overlap of all operating hours from the account, work type, service territory, and service territory member are considered while determining time slots. For more information, see Set Up Operating Hours in Salesforce Scheduler.</td>
</tr>
</tbody>
</table>

• Only the time slots within the period of 31 days from the start date are returned.
Salesforce Scheduler uses multiple factors, such as field values, scheduled appointments, absences, Scheduler Settings, and Scheduling Policies to determine available time slots, including the earliest and latest appointment slots. See How Does Salesforce Scheduler Determine Available Time Slots.

**Note:** If asset scheduling is enabled, you can provide an asset-based service resource in `requiredResourceIds` to retrieve available timeslots for the asset resource.

### Namespace

`LxScheduler`

### Usage

The constructor for this class can’t be called directly. Create an instance of this class using the `GetAppointmentSlotsInputBuilder.build()` method.

This example shows how to get a list of available time slots based on `workTypeGroupId`:

```java
//Build input for GetAppointmentSlots API
lxscheduler.GetAppointmentSlotsInput input = new lxscheduler.GetAppointmentSlotsInputBuilder()
    .setWorkTypeGroupId('0VSxx0000004C92GAE')
    .setTerritoryIds(new List<String>{'0Hhx0000004C92CAE'})
    .setStartTime(System.now().format('yyyy-MM-dd\"T\"HH:mm:ssZ'))
    .setEndTime(System.now().addDays(1).format('yyyy-MM-dd\"T\"HH:mm:ssZ'))
    .setAccountId('001xx000003GYK0AAO')
    .setRequiredResourceIds(new List<String>{'0Hnx0000004C92CAE'})
    .setSchedulingPolicyId('0Vrxx0000004C92CAE')
    .setApiVersion(Double.valueOf('48.0'))
    .build();

String response = lxscheduler.SchedulerResources.getAppointmentSlots(input);
```

This example shows how to get a list of available time slots based on `workType`:

```java
//Build WorkType
lxscheduler.WorkType workType = new lxscheduler.WorkTypeBuilder()
    .setId('08qxx0000004C92AAE')
    .build();

lxscheduler.GetAppointmentSlotsInput input = new lxscheduler.GetAppointmentSlotsInputBuilder()
    .setWorkType(workType)
    .setTerritoryIds(new List<String>{'0Hhx0000004C92CAE'})
    .setStartTime(System.now().format('yyyy-MM-dd\"T\"HH:mm:ssZ'))
    .setEndTime(System.now().addDays(1).format('yyyy-MM-dd\"T\"HH:mm:ssZ'))
    .setAccountId('001xx000003GYK0AAO')
    .setRequiredResourceIds(new List<String>{'0Hnx0000004C92CAE'})
    .setSchedulingPolicyId('0Vrxx0000004C92CAE')
    .setApiVersion(Double.valueOf('48.0'))
    .build();

String response = lxscheduler.SchedulerResources.getAppointmentSlots(input);
```
This example shows how to get a list of available time slots based on `durationInMinutes` and without `workTypeGroupId` or `workType` fields:

```java
//Build WorkType
lxscheduler.WorkType workType = new lxscheduler.WorkTypeBuilder()
    .setDurationInMinutes(60)
    .build();

lxscheduler.GetAppointmentSlotsInput input = new
lxscheduler.GetAppointmentSlotsInputBuilder()
    .setWorkType(workType)
    .setTerritoryIds(new List<String>{'0Hhxx0000004C92CAE'})
    .setRequiredResourceIds(new List<String>{'0Hnxx0000004C92CAE'})
    .setApiVersion(Double.valueOf('48.0'))
    .build();

String response = lxscheduler.SchedulerResources.getAppointmentSlots(input);
```

This example shows a sample response of a list of available time slots:

```json
[{
    "territoryId": "0Hhxx0000004C92CAE",
    "startTime": "2021-02-10T16:00:00.000+0000",
    "endTime": "2021-02-10T16:15:00.000+0000",
    "remainingAppointments": 1
}, {
    "territoryId": "0Hhxx0000004C92CAE",
    "startTime": "2021-02-10T16:15:00.000+0000",
    "endTime": "2021-02-10T16:30:00.000+0000",
    "remainingAppointments": 1
},
]```

## GetAppointmentSlotsInputBuilder Class

Contains methods to build an instance of the `lxscheduler.GetAppointmentSlotsInput` class. A Builder object is obtained by invoking one of the `GetAppointmentSlotsInputBuilder` methods defined by the `GetAppointmentSlotsInput` class.

### Namespace

LxScheduler

### IN THIS SECTION:

GetAppointmentSlotsInputBuilder Methods

### GetAppointmentSlotsInputBuilder Methods

The following are methods for `GetAppointmentSlotsInputBuilder`.
IN THIS SECTION:

build()
Returns an instance of the lxscheduler.GetAppointmentSlotsInput object.

setAccountId(accountId)
Sets the ID of the associated account for which you want to create appointments.

setAllowConcurrentScheduling(allowConcurrentScheduling)
Allows the scheduling of concurrent appointments.

setApiVersion(apiVersion)
Sets the API version of the business logic for the getAppointmentSlots method.

setCorrelationId(correlationId)
Sets the correlation ID.

setEndTime(endTime)
Sets the scheduling end time.

setEngagementChannelTypeIds(engagementChannelTypeIds)
Sets an engagement channel type.

setPrimaryResourceId(primaryResourceId)
Sets the ID of the primary resource.

setRequiredResourceIds(requiredResourceIds)
Sets the resource IDs.

setSchedulingPolicyId(schedulingPolicyId)
Sets the ID of the AppointmentSchedulingPolicy object.

setStartTime(startTime)
Sets the scheduling start time.

setTerritoryIds(territoryIds)
Sets the IDs of service territories.

setWorkType(workType)
Sets the type of work to be performed.

setWorkTypeGroupId(workTypeGroupId)
Sets the ID of the work type group.

**build()**

Returns an instance of the lxscheduler.GetAppointmentSlotsInput object.

**Signature**

```java
public lxscheduler.GetAppointmentSlotsInput build()
```

**Return Value**

Type: lxscheduler.GetAppointmentSlotsInput
**setAccountId(accountId)**
Sets the ID of the associated account for which you want to create appointments.

**Signature**

```java
public lxscheduler.GetAppointmentSlotsInputBuilder setAccountId(String accountId)
```

**Parameters**

- **accountId**
  - **Type:** String
  - The ID of the associated account.

**Return Value**

- **Type:** lxscheduler.GetAppointmentSlotsInputBuilder

**setAllowConcurrentScheduling(allowConcurrentScheduling)**
Allows the scheduling of concurrent appointments.

**Signature**

```java
public lxscheduler.GetAppointmentSlotsInputBuilder setAllowConcurrentScheduling(Boolean allowConcurrentScheduling)
```

**Parameters**

- **allowConcurrentScheduling**
  - **Type:** Boolean
  - If true, allows scheduling of concurrent appointments in a time slot. If false, concurrent appointments are not allowed. The default is false. Available in API version 47.0 and later.

**Return Value**

- **Type:** lxscheduler.GetAppointmentSlotsInputBuilder

**setApiVersion(apiVersion)**
Sets the API version of the business logic for the getAppointmentSlots method.

**Signature**

```java
public lxscheduler.GetAppointmentSlotsInputBuilder setApiVersion(Double apiVersion)
```

**Parameters**

- **apiVersion**
  - **Type:** Double
Usage
The specified parameter must use the correct API version. For example, if API version is set to 45.0 and `primaryResourceId` is set (which is available in API version 48.0 and later), then this field is ignored. If no API version or incorrect API version is passed in the request body, by default the latest version is used.

Note: The API is available since version 45.0.

Return Value
Type: lxscheduler.GetAppointmentSlotsInputBuilder

`setCorrelationId(correlationId)`
Sets the correlation ID.

Signature
`public lxscheduler.GetAppointmentSlotsInputBuilder setCorrelationId(String correlationId)`

Parameters
`correlationId`
Type: String
ID to pass custom information to the `ServiceResourceScheduleHandler` Apex interface. For example, you can use the correlation ID to identify the app, website, or any other external system that calls this Apex interface implementation. If you don’t pass a custom value, a randomly generated identifier is passed. Available in API version 53.0 and later.

Return Value
Type: lxscheduler.GetAppointmentSlotsInputBuilder

`setEndTime(endTime)`
Sets the scheduling end time.

Signature
`public lxscheduler.GetAppointmentSlotsInputBuilder setEndTime(String endTime)`

Parameters
`endTime`
Type: String
The latest time that a time slot can end (inclusive). If end time is not specified, it defaults to 31 days.

Usage
The specified string should use the standard date format `"[yyyy-MM-dd'T'HH:mm:ssZ]"` in the local time zone. Defaults to the user’s time zone.
Return Value
Type: lxscheduler.GetAppointmentSlotsInputBuilder

**setEngagementChannelTypeIds** (engagementChannelTypeIds)
Sets an engagement channel type.

Signature
```
public lxscheduler.GetAppointmentSlotsInputBuilder
setEngagementChannelTypeIds(List<String> engagementChannelTypeIds)
```

Parameters
```
engagementChannelTypeIds
    Type: List<String>
```
The ID of the engagement channel type record. The availability of time slots is filtered based on the engagement channel type selected. This field is available in API version 56.0 and later.

Note: This field supports only one engagement channel type ID.

Return Value
Type: lxscheduler.GetAppointmentSlotsInputBuilder

Usage
You can use engagement channel types only in these cases:
- The Schedule Appointments Using Engagement Channels setting is enabled in Salesforce Scheduler Settings in your Salesforce org.
- Shifts are defined in the scheduling policy. For more information on setting up shifts in scheduling policy, see Define Shift Rules in Scheduling Policy.

Note: Engagement channel types are not supported with operating-hours rules in the scheduling policy.

**setPrimaryResourceId** (primaryResourceId)
Sets the ID of the primary resource.

Signature
```
public lxscheduler.GetAppointmentSlotsInputBuilder
setPrimaryResourceId(String primaryResourceId)
```

Parameters
```
primaryResourceId
    Type: String
```
The ID of the primary resource in multi-resource scheduling. Required only when multi-resource scheduling is enabled. Available in API version 48.0 and later.

**Return Value**
Type: lxscheduler.GetAppointmentSlotsInputBuilder

**setRequiredResourceIds(requiredResourceIds)**
Sets the resource IDs.

**Signature**
```java
public lxscheduler.GetAppointmentSlotsInputBuilder setRequiredResourceIds(List<String> requiredResourceIds)
```

**Parameters**

- `requiredResourceIds`
  Type: List<String>
  List of resource IDs that must be available during the time slot. This is a required field.

**Return Value**
Type: lxscheduler.GetAppointmentSlotsInputBuilder

**setSchedulingPolicyId(schedulingPolicyId)**
Sets the ID of the AppointmentSchedulingPolicy object.

**Signature**
```java
public lxscheduler.GetAppointmentSlotsInputBuilder setSchedulingPolicyId(String schedulingPolicyId)
```

**Parameters**

- `schedulingPolicyId`
  Type: String
  If no scheduling policy is passed in the request body, the default configurations are used.

**Return Value**
Type: lxscheduler.GetAppointmentSlotsInputBuilder

**setStartTime(startTime)**
Sets the scheduling start time.
Signature

```java
public lxscheduler.GetAppointmentSlotsInputBuilder setStartTime(String startTime)
```

Parameters

```java
startTime
Type: String
```

The earliest time that a time slot can begin (inclusive). Defaults to the current time of the request, if empty.

Usage

The specified string should use the standard date format "T\[yyyy-MM-dd'T'HH:mm:ssZ\]" in the local time zone. Defaults to the user's time zone.

Return Value

Type: lxscheduler.GetAppointmentSlotsInputBuilder

```java
setTerritoryIds(territoryIds)
```

Sets the IDs of service territories.

Signature

```java
public lxscheduler.GetAppointmentSlotsInputBuilder setTerritoryIds(List<String> territoryIds)
```

Parameters

```java
territoryIds
Type: List<String>
```

List of IDs of service territories, where the work that is being requested is performed. This is a required field.

Return Value

Type: lxscheduler.GetAppointmentSlotsInputBuilder

```java
setWorkType(workType)
```

Sets the type of work to be performed.

Signature

```java
public lxscheduler.GetAppointmentSlotsInputBuilder setWorkType(lxscheduler.WorkType workType)
```

Parameters

```java
workType
Type: lxscheduler.WorkType
```
This method takes input as an instance of the `lxscheduler.WorkType` class. Build the instance of the input class using the `lxscheduler.WorkTypeBuilder` class.

Required if `workTypeGroupId` is not given.

**Return Value**

Type: `lxscheduler.GetAppointmentSlotsInputBuilder`

**setWorkTypeGroupId(workTypeGroupId)**

Sets the ID of the work type group.

**Signature**

```java
public lxscheduler.GetAppointmentSlotsInputBuilder setWorkTypeGroupId(String workTypeGroupId)
```

**Parameters**

- `workTypeGroupId`
  
  Type: `String`
  
  The ID of the work type group containing the work types that are being performed.

**Return Value**

Type: `lxscheduler.GetAppointmentSlotsInputBuilder`

---

**SchedulerResources Class**

Contains methods that holds the business logic to get resources availability.

**Namespace**

LxScheduler

**Implementation Considerations**

Apex implementation of the methods in the `SchedulerResources` class should adhere to Apex Governor Limits. It includes synchronous heap size limit, synchronous CPU time limit, and synchronous concurrent transactions for long running transactions. To avoid governor limits, you must tune the input by reducing the time frame, limiting number of service resources, or limiting number or territories at a time. This will reduce the overall transaction time and response size of the implementation. For more information on standard Apex Governor Limits, see Salesforce Developer Limits and Allocations Quick Reference.

**Example**

To get list of available service resources (appointment candidates):

```java
String response = lxscheduler.SchedulerResources.getAppointmentCandidates(input);
```
To get a list of available appointment time slots for a resource:

```java
String response = lxscheduler.SchedulerResources.getAppointmentSlots(input);
```

IN THIS SECTION:
- **SchedulerResources Methods**

**SchedulerResources Methods**

The following are methods for SchedulerResources.

IN THIS SECTION:
- `getAppointmentCandidates(getAppointmentCandidatesInput)`
  Returns a list of service resources based on work type group or work type and service territories.
- `getAppointmentSlots(getAppointmentSlotsInput)`
  Returns a list of available appointment time slots for a resource based on given work type group or work type and service territories.
- `setAppointmentCandidatesMock(expectedResponse)`
  Sets a mock object when running tests for the `getAppointmentCandidates` method.
- `setAppointmentSlotsMock(expectedResponse)`
  Sets a mock object when running tests for the `getAppointmentSlots` method.

**getAppointmentCandidates (getAppointmentCandidatesInput)**

Returns a list of service resources based on work type group or work type and service territories.

Set up Salesforce Scheduler before making requests. This setup includes creating or configuring Service Resources, Service Territory Members, Work Type Groups, Work Types, Work Type Group Members, and Service Territory Work Types. See Set Up Salesforce Scheduler for more information.

The appointment time slots are determined based on multiple factors, such as field values, scheduled appointments, absences, Scheduler Settings, and Scheduling Policies to determine available time slots. See How Salesforce Scheduler Determines Available Time Slots for more information.

The following factors are considered for returning start time and end time of resources.

**Resource Availability**
- Determined using service territory member, service territory, work type, and account operating hours fields.

**Resource Unavailability**
- Determined by resource absences, existing appointments that the resource is assigned to. The resource must be marked as a required resource for the appointment with a status that isn’t in closed, canceled, or completed.

**Appointment Start Time Interval in the Scheduling Policy**
- Appointment start time interval field in the Scheduling Policy is used to determine when the appointment can start. This interval can be 5, 10, 15, 20, 30, or 60. By default, it’s set to 15.

**Work Type Duration**
- The end time is calculated as start time + duration of the work type.

**Note:** If asset scheduling is enabled, the response also includes asset-based candidates.
Signature

```java
public static String getAppointmentCandidates(lxscheduler.GetAppointmentCandidatesInput getAppointmentCandidatesInput)
```

Parameters

`getAppointmentCandidatesInput`
Type: `lxscheduler.GetAppointmentCandidatesInput`

This method takes input as an instance of the `lxscheduler.GetAppointmentCandidatesInput` class. Build the instance of the input class using the `lxscheduler.GetAppointmentCandidatesInputBuilder` class.

Return Value
Type: `String`

`getAppointmentSlots(getAppointmentSlotsInput)`

Returns a list of available appointment time slots for a resource based on given work type group or work type and service territories. The appointment time slots are determined based on your Salesforce Scheduler data model configurations. Here are some prerequisites that you can consider while setting up data.

- Set up Salesforce Scheduler before making your requests. The setup includes creating or configuring Service Resources, Service Territory Members, Work Type Groups, Work Types, Work Type Group Members, and Service Territory Work Types. See Manage Business Information in Salesforce Scheduler for more information.
- Configure a work type mapped for each territory in the request body via Service Territory Work Type. Map the same work type to the work type group, via work type group member.

The following factors affect how time slots are calculated and returned.

- Timezones that differ across operating hours are handled and results are always returned in UTC.
- The resource must be marked as a required resource on the assigned resource object.
- The resource is considered unavailable if the status categories of the resource assigned to service appointments are other than Canceled, Cannot Complete, and Completed.
- Resource Absences of all types are considered unavailable from start to end.
- The following fields of Work Type records, if configured, are used to fine-tune time slot requirements. For more information, see Create Work Types in Salesforce Scheduler.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Timeframe Start</td>
<td>Time slots sooner than current time + <strong>Timeframe Start</strong> aren’t returned.</td>
</tr>
<tr>
<td>Timeframe End</td>
<td>Time slots later than current time + <strong>Timeframe End</strong> aren’t returned.</td>
</tr>
<tr>
<td>Block Time Before Appointment</td>
<td>The time period before the appointment is considered as unavailable.</td>
</tr>
<tr>
<td>Block Time After Appointment</td>
<td>The time period after the appointment is considered as unavailable.</td>
</tr>
<tr>
<td>Operating Hours</td>
<td>The overlap of all operating hours from the account, work type, service territory, and service territory member are considered while determining time slots. For more information, see <strong>Set Up Operating Hours in Salesforce Scheduler.</strong></td>
</tr>
</tbody>
</table>
Only the time slots within the period of 31 days from the start date are returned.

Salesforce Scheduler uses multiple factors, such as field values, scheduled appointments, absences, Scheduler Settings, and Scheduling Policies to determine available time slots, including the earliest and latest appointment slots. See How Does Salesforce Scheduler Determine Available Time Slots.

Note: If asset scheduling is enabled, you can provide an asset-based service resource in requiredResourceIds to retrieve available timeslots for the asset resource.

Signature

```
public static String getAppointmentSlots(lxscheduler.GetAppointmentSlotsInput getAppointmentSlotsInput)
```

Parameters

```
getAppointmentSlotsInput
Type: lxscheduler.GetAppointmentSlotsInput
```

This method takes input as an instance of the lxscheduler.GetAppointmentSlotsInput class. Build the instance of the input class using the lxscheduler.GetAppointmentSlotsInputBuilder class.

Return Value

Type: String

```
setAppointmentCandidatesMock(expectedResponse)
```

Sets a mock object when running tests for the getAppointmentCandidates method.

This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

Signature

```
public static void setAppointmentCandidatesMock(String expectedResponse)
```

Parameters

```
expectedResponse
Type: String
```

Return Value

Type: void

This example shows a sample implementation of the GetAppointmentCandidates class:

```
public class AppointmentCandidateService {
    //Instance members for parsing
    public String startTime;
    public String endTime;
    public List<String> resources;
    public String territoryId;
```
public static List<AppointmentCandidateService> getAppointmentCandidates() {
    // Build input for GetAppointmentCandidates API
    lxscheduler.GetAppointmentCandidatesInput input = new lxscheduler.GetAppointmentCandidatesInputBuilder()
        .setWorkTypeGroupId('0VSRM0000000AGT4A2')
        .setTerritoryIds(new List<String>{'0HhRM0000000G8W0AU'})
        .setStartTime(System.now().format('yyyy-MM-dd\"T\"HH:mm:ssZ','America/Los_Angeles'))
        .setEndTime(System.now().addDays(2).format('yyyy-MM-dd\"T\"HH:mm:ssZ','America/Los_Angeles'))
        .setSchedulingPolicyId('0VrRM00000000D0')
        .setApiVersion(Double.valueOf('50.0'))
        .build();
    List<AppointmentCandidateService> vList = parse(lxscheduler.SchedulerResources.getAppointmentCandidates(input));
    return vList;
}

private static List<AppointmentCandidateService> parse(String json) {
    return (List<AppointmentCandidateService>) System.JSON.deserialize(json, List<AppointmentCandidateService>.class);
}

This example shows how to set a sample mock using the setAppointmentCandidatesMock method:

@isTest
private class GetAppointmentCandidatesTest {
    static testMethod void getAppCandidatesTest() {
        String expectedResponse = ' [
            { "startTime": "2021-03-18T16:00:00.000+0000", ' +
            "endTime": "2021-03-18T17:00:00.000+0000", ' +
            "resources": [ ' +
                "0HnRM0000000Fxv0AE"' +
            ], ' +
            "territoryId": "0HhRM0000000G8W0AU"' +
            ' }, ' +
            { "startTime": "2021-03-18T19:00:00.000+0000", ' +
            "endTime": "2021-03-18T20:00:00.000+0000", ' +
            "resources": [ ' +
                "0HnRM0000000Fxv0AE"' +
            ], ' +
            "territoryId": "0HhRM0000000G8W0AU"' +
            ' }];
        lxscheduler.SchedulerResources.setAppointmentCandidatesMock(expectedResponse);
        Test.startTest();
    }
}
List<AppointmentCandidateService> candidateList = AppointmentCandidateService.getAppointmentCandidates();
    System.assertEquals(2, candidateList.size(), 'Should return only 2 records!');
    Test.stopTest();
}

**setAppointmentSlotsMock(expectedResponse)**

Sets a mock object when running tests for the `getAppointmentSlots` method.
This constructor is intended for test usage and throws an exception if used outside of the Apex test context.

**Signature**

```java
public static void setAppointmentSlotsMock(String expectedResponse)
```

**Parameters**

- **expectedResponse**
  - Type: `String`

**Return Value**

- Type: `void`

**SkillRequirement Class**

Contains information about the set of skills that are required to complete a particular task for a work type.

**Namespace**

`LxScheduler`

**Usage**

The constructor for this class can't be called directly. Create an instance of this class using the `SkillRequirementBuilder.build()` method.

**SkillRequirementBuilder Class**

Contains methods to build an instance of the `lxscheduler.SkillRequirement` class.
A `Builder` object is obtained by invoking one of the `SkillRequirementBuilder` methods defined by the `SkillRequirement` class.

**Namespace**

`LxScheduler`
IN THIS SECTION:
  SkillRequirementBuilder Methods

**SkillRequirementBuilder Methods**

The following are methods for `SkillRequirementBuilder`.

**IN THIS SECTION:**
  build()
  Returns an instance of the `lxscheduler.SkillRequirement` object.

  setSkillId(skillId)
  Sets the skill that is required to complete a particular task for a work type. This is a required field.

  setSkillLevel(skillLevel)
  Sets the level of the skill that is required to complete a particular task for a work type

### build()

Returns an instance of the `lxscheduler.SkillRequirement` object.

**Signature**

```
public lxscheduler.SkillRequirement build()
```

**Return Value**

Type: `lxscheduler.SkillRequirement`

### setSkillId(skillId)

Sets the skill that is required to complete a particular task for a work type. This is a required field.

**Signature**

```
public lxscheduler.SkillRequirementBuilder setSkillId(String skillId)
```

**Parameters**

- `skillId`
  Type: `String`

**Return Value**

Type: `lxscheduler.SkillRequirementBuilder`

### setSkillLevel(skillLevel)

Sets the level of the skill that is required to complete a particular task for a work type.
Signature

```java
public lxscheduler.SkillRequirementBuilder setSkillLevel(Double skillLevel)
```

Parameters

`skillLevel`

Type: `Double`

The skill levels can range from zero to 99.99. Depending on your business needs, you might want the skill level to reflect years of experience, certification levels, or license classes.

Return Value

Type: `lxscheduler.SkillRequirementBuilder`

---

**WorkType Class**

Contains information about the type of work to be performed.

**Namespace**

`LxScheduler`

**Usage**

The constructor for this class can't be called directly. Create an instance of this class using the `WorkTypeBuilder.build()` method.

---

**WorkTypeBuilder Class**

Contains methods to build an instance of the `lxscheduler.WorkType` class.

A Builder object is obtained by invoking one of the `WorkTypeBuilder` methods defined by the `WorkType` class.

**Namespace**

`LxScheduler`

---

**WorkTypeBuilder Methods**

The following are methods for `WorkTypeBuilder`.

**IN THIS SECTION:**

- `build()`

  Returns an instance of the `lxscheduler.WorkType` object.
**setBlockTimeAfterAppointmentInMinutes(blockTimeAfterAppointmentInMinutes)**
Sets the time period, in minutes.

**setBlockTimeBeforeAppointmentInMinutes(blockTimeBeforeAppointmentInMinutes)**
Sets the time period, in minutes.

**setDurationInMinutes(durationInMinutes)**
Sets the event length.

**setId(id)**
Sets the ID of the work type to the specified ID.

**setOperatingHoursId(operatingHoursId)**
Sets the overlap of operating hours.

**setSkillRequirements(skillRequirements)**
Sets the skills that are required to complete a particular task for a work type.

**setTimeFrameEndInMinutes(timeFrameEndInMinutes)**
Sets the end of the timeframe.

**setTimeFrameStartInMinutes(timeFrameStartInMinutes)**
Sets the beginning of the timeframe.

**build()**
Returns an instance of the lxscheduler.WorkType object.

**Signature**

```java
public lxscheduler.WorkType build()
```

**Return Value**

Type: lxscheduler.WorkType

**setBlockTimeAfterAppointmentInMinutes(blockTimeAfterAppointmentInMinutes)**
Sets the time period, in minutes.

**Signature**

```java
public lxscheduler.WorkTypeBuilder setBlockTimeAfterAppointmentInMinutes(Integer blockTimeAfterAppointmentInMinutes)
```

**Parameters**

- **blockTimeAfterAppointmentInMinutes**
  - Type: `Integer`
  - The time period after the appointment is considered unavailable.
Return Value
Type: lxscheduler.WorkTypeBuilder

**setBlockTimeBeforeAppointmentInMinutes** *(blockTimeBeforeAppointmentInMinutes)*
Sets the time period, in minutes.

**Signature**
```
public lxscheduler.WorkTypeBuilder setBlockTimeBeforeAppointmentInMinutes(Integer blockTimeBeforeAppointmentInMinutes)
```

**Parameters**
- **blockTimeBeforeAppointmentInMinutes**
  Type: Integer
  The time period before the appointment is considered as unavailable.

Return Value
Type: lxscheduler.WorkTypeBuilder

**setDurationInMinutes** *(durationInMinutes)*
Sets the event length.

**Signature**
```
public lxscheduler.WorkTypeBuilder setDurationInMinutes(Integer durationInMinutes)
```

**Parameters**
- **durationInMinutes**
  Type: Integer
  Contains the event length, in minutes. Required if **id** is not given.

Return Value
Type: lxscheduler.WorkTypeBuilder

**setId** *(id)*
Sets the ID of the work type to the specified ID.

**Signature**
```
public lxscheduler.WorkTypeBuilder setId(String id)
```
Parameters

id
   Type: String
   The ID of the work type. Required if durationInMinutes is not given.

Return Value

Type: lx scheduler.WorkTypeBuilder

setOperatingHoursId(operatingHoursId)

Sets the overlap of operating hours.

Signature

public lx scheduler.WorkTypeBuilder setOperatingHoursId(String operatingHoursId)

Parameters

operatingHoursId
   Type: String
   The overlap of all operating hours from the account, work type, service territory, and service territory member are considered while determining time slots.

Return Value

Type: lx scheduler.WorkTypeBuilder

setSkillRequirements(skillRequirements)

Sets the skills that are required to complete a particular task for a work type.

Signature

public lx scheduler.WorkTypeBuilder
setSkillRequirements(List<lx scheduler.SkillRequirement> skillRequirements)

Parameters

skillRequirements
   Type: List<lx scheduler.SkillRequirement>
   This method takes input as an instance of the lx scheduler.SkillRequirement class. Build the instance of the input class using the lx scheduler.SkillRequirementBuilder class.

Return Value

Type: lx scheduler.WorkTypeBuilder
**setTimeFrameEndInMinutes (timeFrameEndInMinutes)**

Sets the end of the timeframe.

**Signature**

```java
public lxscheduler.WorkTypeBuilder setTimeFrameEndInMinutes(Integer timeFrameEndInMinutes)
```

**Parameters**

- `timeFrameEndInMinutes`  
  Type: `Integer`

**Return Value**

Type: `lxscheduler.WorkTypeBuilder`

**setTimeFrameStartInMinutes (timeFrameStartInMinutes)**

Sets the beginning of the timeframe.

**Signature**

```java
public lxscheduler.WorkTypeBuilder setTimeFrameStartInMinutes(Integer timeFrameStartInMinutes)
```

**Parameters**

- `timeFrameStartInMinutes`  
  Type: `Integer`

**Return Value**

Type: `lxscheduler.WorkTypeBuilder`

**ServiceResourceScheduleHandler Interface**

Allows an implementing class to check external calendar events to find already booked time slots for the requested service resources. This interface is part of Salesforce Scheduler.

**Namespace**

`LxScheduler`

**Usage**

The `lxscheduler.ServiceResourceScheduleHandler` interface is called by Salesforce Scheduler APIs.
To implement this interface, you must first declare a class with the `implements` keyword as follows:

```java
public class ServiceResourceScheduleHandlerImpl implements LxScheduler.ServiceResourceScheduleHandler{
}
```

Next, your class must provide an implementation for the following method:

```java
public static List<LxScheduler.ServiceResourceSchedule> getUnavailableTimeslots(LxScheduler.ServiceAppointmentRequestInfo requestInfo){
    //Your code here
}
```

The implemented method must be declared as `global` or `public`.

IN THIS SECTION:
- **ServiceResourceScheduleHandler Methods**
- **ServiceResourceScheduleHandler Example Implementation**

**ServiceResourceScheduleHandler Methods**

The following are methods for `ServiceResourceScheduleHandler`.

IN THIS SECTION:
- `getUnavailableTimeslots(var1)`

`getUnavailableTimeslots(var1)`

Passes the required information to get unavailable time slots from an external system. The implementation of this method returns the `lxscheduler.ServiceResourceSchedule` class.

**Signature**

```java
public List<lxscheduler.ServiceResourceSchedule> getUnavailableTimeslots(LxScheduler.ServiceAppointmentRequestInfo var1)
```

**Parameters**

`var1`

Type: `lxscheduler.ServiceAppointmentRequestInfo`

Represents the list of parameters that are passed to the `ServiceResourceScheduleHandler` interface.

**Return Value**

Type: `List<lxscheduler.ServiceResourceSchedule>`
ServiceResourceScheduleHandler Example Implementation

This is an example implementation of the `lxscheduler.ServiceResourceScheduleHandler` interface.

```java
/**
 * Implement interface `lxscheduler.ServiceResourceScheduleHandler`
 * This class is called when fetching service resources and time slots through Salesforce Scheduler API.*/

public class ServiceResourceScheduleHandlerImpl implements lxscheduler.ServiceResourceScheduleHandler {
    // The main interface method.
    public static List<lxscheduler.ServiceResourceSchedule> getUnavailableTimeslots(lxscheduler.ServiceAppointmentRequestInfo requestInfo) {
        // Request info values.
        List<lxscheduler.ServiceResourceInfo> serviceResources = requestInfo.getServiceResources();
        DateTime startDate = requestInfo.getStartDate();
        DateTime endDate = requestInfo.getEndDate();

        Set<lxscheduler.UnavailableTimeslot> unavailabilityIntervals = new Set<lxscheduler.UnavailableTimeslot>();

        // This is a dummy response. Implement your own business logic to connect to your internal or external systems.
        for (Integer i = 0; i < 5; i++) {
            // Set the unavailability intervals of a service resource.
            unavailabilityIntervals.add(new lxscheduler.UnavailableTimeslot(startDate.addMinutes(15*i), startDate.addMinutes(15*(i+1))));
        }

        for (lxscheduler.ServiceResourceInfo ServiceResource : serviceResources) {
            // Set the unavailability of Service resource.
            resourceUnavailability.add(new lxscheduler.ServiceResourceSchedule(ServiceResource.getServiceResourceId(), unavailabilityIntervals));
        }

        return resourceUnavailability;
    }
}
```

This example shows how to set a sample test mock using the `lxscheduler.ServiceResourceScheduleHandler` interface.

```java
@isTest
private class ServiceResourceScheduleHandlerImplTest {
    static testMethod void getUnavailableTimeslotsTest() {
        // Initializing the test execution with mock values. Change it according to the
```
implementation.
// In case of non-test execution, the lxscheduler.ServiceAppointmentRequestInfo instance will automatically initialize.

// Mock values for lxscheduler.ServiceResourceInfo
String userId = '05D2000000I1N6IAK';
String userName = 'someuser@example.com';
String email = 'someuser@example.com';
String serviceResourceId = '0HnD20000004C9bKAE';
List<String> territoryIds = new List<String>();
String resourceType = 'T';
lxscheduler.ServiceResourceInfo serviceResInfo = new lxscheduler.ServiceResourceInfo(userId, userName, email, serviceResourceId, territoryIds, resourceType);

// Mock values for lxscheduler.ServiceAppointmentRequestInfo
DateTime startDate = System.now();
DateTime endDate = System.now();
List<lxscheduler.ServiceResourceInfo> serviceResources = new List<lxscheduler.ServiceResourceInfo>();
serviceResources.add(serviceResInfo);
String schedulingPolicyId = '0VrD20000004C9S';
String workTypeGroupId = '0VSD20000004C93OA';
String accountId = '001D2000002pkXwIAI';
String primaryResourceId = '0HnD20000004C9bKAE';
String workTypeId = '08qD20000004C9XIUA';
String correlationId = 'SOME_ID';
lxscheduler.ServiceAppointmentRequestInfo mockRequestInfo = new lxscheduler.ServiceAppointmentRequestInfo(startDate, endDate, serviceResources, schedulingPolicyId, workTypeGroupId, accountId, primaryResourceId, workTypeId, correlationId);

ServiceResourceScheduleHandlerImpl.getUnavailableTimeslots(mockRequestInfo);
}

---

**ServiceAppointmentRequestInfo Class**

Represents the list of parameters that are passed to the ServiceResourceScheduleHandler interface. This class is implemented internally by Apex.

**Namespace**

LxScheduler

IN THIS SECTION:

- ServiceAppointmentRequestInfo Constructors
ServiceAppointmentRequestInfo Methods

ServiceAppointmentRequestInfo Constructors

The following are constructors for ServiceAppointmentRequestInfo.

IN THIS SECTION:

ServiceAppointmentRequestInfo(startDate, endDate, ServiceResources, SchedulingPolicyId, workTypeGroupId, accountId, primaryResourceId, workTypeId, correlationId)

Creates a new instance of the lxscheduler.ServiceAppointmentRequestInfo class using the specified start date, end date, service resources, scheduling policy, work type group, account ID, primary resource, work type, and correlation.

ServiceAppointmentRequestInfo(startDate, endDate, ServiceResources, SchedulingPolicyId, workTypeGroupId, accountId, primaryResourceId, workTypeId, correlationId)

Creates a new instance of the lxscheduler.ServiceAppointmentRequestInfo class using the specified start date, end date, service resources, scheduling policy, work type group, account ID, primary resource, work type, and correlation.

Signature

public ServiceAppointmentRequestInfo(Datetime startDate, Datetime endDate, List<lxscheduler.ServiceResourceInfo> ServiceResources, String SchedulingPolicyId, String workTypeGroupId, String accountId, String primaryResourceId, String workTypeId, String correlationId)

Parameters

startDate
Type: Datetime
The start date and time for which unavailable time slots are requested.

date
Type: Datetime
The end date and time for which unavailable time slots are requested.

ServiceResources
Type: List<lxscheduler.ServiceResourceInfo>
The list of requested service resources for the unavailable time slots.

SchedulingPolicyId
Type: String
The ID of the scheduling policy.

workTypeGroupId
Type: String
The work type group ID.

accountId
Type: String
The account ID of an existing user.

`primaryResourceId`
Type: String
The ID of the primary service resource.

`workTypeId`
Type: String
The work type ID.

`correlationId`
Type: String
A unique identifier for a service appointment request.

### ServiceAppointmentRequestInfo Methods

The following are methods for `ServiceAppointmentRequestInfo`.

**IN THIS SECTION:**
- `getAccountId()`: Returns the account ID of the customer if the API request contains one.
- `getCorrelationId()`: Returns a unique identifier for a request.
- `getEndDate()`: Returns the end date and time for which unavailable time slots are requested.
- `getPrimaryResourceId()`: Returns the ID of the primary service resource.
- `getSchedulingPolicyId()`: Returns the ID of the scheduling policy that the API request contains.
- `getServiceResources()`: Returns the list of requested service resources for the unavailable time slots.
- `getStartDate()`: Returns the start date and time for which unavailable time slots are requested.
- `getWorkTypeGroupId()`: Returns the work type group ID if the API request contains one.
- `getWorkTypeId()`: Returns the work type ID if the API request contains one.

**getAccountId()**

Returns the account ID of the customer if the API request contains one.

**Signature**

```java
public String getAccountId()
```
Return Value
Type: String

getCorrelationId()
Returns a unique identifier for a request.

Signature
public String getCorrelationId()

Return Value
Type: String

gEndDate()
Returns the end date and time for which unavailable time slots are requested.

Signature
public Datetime getEndDate()

Return Value
Type: Datetime

getPrimaryResourceId()
Returns the ID of the primary service resource.

Signature
public String getPrimaryResourceId()

Return Value
Type: String

getSchedulingPolicyId()
Returns the ID of the scheduling policy that the API request contains.

Signature
public String getSchedulingPolicyId()

Return Value
Type: String
**getServiceResources()**
Returns the list of requested service resources for the unavailable time slots.

**Signature**
```java
public List<lxscheduler.ServiceResourceInfo> getServiceResources()
```

**Return Value**
Type: List<lxscheduler.ServiceResourceInfo>

**getStartDate()**
Returns the start date and time for which unavailable time slots are requested.

**Signature**
```java
public Datetime getStartDate()
```

**Return Value**
Type: Datetime

**getWorkTypeGroupId()**
Returns the work type group ID if the API request contains one.

**Signature**
```java
public String getWorkTypeGroupId()
```

**Return Value**
Type: String

**getWorkTypeId()**
Returns the work type ID if the API request contains one.

**Signature**
```java
public String getWorkTypeId()
```

**Return Value**
Type: String

**ServiceResourceInfo Class**
Contains information about a service resource.
Namespace
LxScheduler

IN THIS SECTION:
ServiceResourceInfo Constructors
ServiceResourceInfo Methods

ServiceResourceInfo Constructors
The following are constructors for ServiceResourceInfo.

IN THIS SECTION:
ServiceResourceInfo(userId, userName, email, serviceResourceId, territoryIds, resourceType)
Creates a new instance of the lxscheduler.ServiceResourceInfo class using the specified service resource details.

ServiceResourceInfo(userId, userName, email, serviceResourceId, territoryIds, resourceType)
Creates a new instance of the lxscheduler.ServiceResourceInfo class using the specified service resource details.

Signature
public ServiceResourceInfo(String userId, String userName, String email, String serviceResourceId, List<String> territoryIds, String resourceType)

Parameters
userId
Type: String
The user ID of the service resource.

userName
Type: String
The user name of the service resource.

email
Type: String
The email ID of the service resource.

serviceResourceId
Type: String
The ID of the service resource.

territoryIds
Type: List<String>
A list of requested service territories for the service resource.
resourceType
Type: String
The type of the service resource such as Technician or Asset.

ServiceResourceInfo Methods
The following are methods for ServiceResourceInfo.

IN THIS SECTION:

getEmail()
Returns the email ID of the service resource.
getResourceType()
Returns the type of the service resource such as Technician or Asset.
getServiceResourceId()
Returns the ID of the service resource.
getTerritoryIds()
Returns a list of requested service territories for the service resource.
getUserId()
Returns the user ID of the service resource.
getUserName()
Returns the user name of the service resource.

getEmail()
Returns the email ID of the service resource.

Signature
public String getEmail()

Return Value
Type: String

ggetResourceType()
Returns the type of the service resource such as Technician or Asset.

Signature
public String getResourceType()

Return Value
Type: String
getServiceResourceId()  
Returns the ID of the service resource.

Signature  
public String getServiceResourceId()

Return Value  
Type: String

getTerritoryIds()  
Returns a list of requested service territories for the service resource.

Signature  
public List<String> getTerritoryIds()

Return Value  
Type: List<String>

getUserId()  
Returns the user ID of the service resource.

Signature  
public String getUserId()

Return Value  
Type: String

getUserName()  
Returns the user name of the service resource.

Signature  
public String getUserName()

Return Value  
Type: String

ServiceResourceSchedule Class  
Use this class to pass results from your implemented Apex class to the ServiceResourceScheduleHandler interface methods.
Namespace
LxScheduler

IN THIS SECTION:
ServiceResourceSchedule Constructors
ServiceResourceSchedule Properties

ServiceResourceSchedule Constructors
The following are constructors for ServiceResourceSchedule.

IN THIS SECTION:
ServiceResourceSchedule(serviceResourceId, unavailableTimeslots)
Creates a new instance of lxscheduler.ServiceResourceSchedule class.

ServiceResourceSchedule(serviceResourceId, unavailableTimeslots)

Signature
public ServiceResourceSchedule(String serviceResourceId,
Set<lxscheduler.UnavailableTimeslot> unavailableTimeslots)

Parameters
serviceResourceId
Type: String
Record ID of the service resource.

unavailableTimeslots
Type: Set<lxscheduler.UnavailableTimeslot>
An instance of lxscheduler.UnavailableTimeslot class.

ServiceResourceSchedule Properties
The following are properties for ServiceResourceSchedule.

IN THIS SECTION:
serviceResourceId
Record ID of the service resource.

unavailableTimeslots
An instance of lxscheduler.UnavailableTimeslot class.
**serviceResourceId**
Record ID of the service resource.

**Signature**
```
public String serviceResourceId {get; set;}
```

**Property Value**
Type: String

**unavailableTimeslots**
An instance of lxscheduler.UnavailableTimeslot class.

**Signature**
```
public Set<lxscheduler.UnavailableTimeslot> unavailableTimeslots {get; set;}
```

**Property Value**
Type: Set<lxscheduler.UnavailableTimeslot>

### UnavailableTimeslot Class
Use this class to pass the unavailable time slots to the lxscheduler.ServiceResourceSchedule class.

### Namespace
LxScheduler

**IN THIS SECTION:**

- UnavailableTimeslot Constructors
- UnavailableTimeslot Properties

### UnavailableTimeslot Constructors
The following are constructors for UnavailableTimeslot.

**IN THIS SECTION:**

- UnavailableTimeslot(timeMin, timeMax)
  Creates an instance of lxscheduler.UnavailableTimeslot class.

**UnavailableTimeslot(timeMin, timeMax)**
Creates an instance of lxscheduler.UnavailableTimeslot class.
Signature

```java
public UnavailableTimeslot(Datetime timeMin, Datetime timeMax)
```

Parameters

- `timeMin`
  Type: `Datetime`
  Start time of an unavailable time slot.

- `timeMax`
  Type: `Datetime`
  End time of an unavailable time slot.

UnavailableTimeslot Properties

The following are properties for `UnavailableTimeslot`.

**IN THIS SECTION:**

- `timeMax`
  End time of an unavailable time slot.

- `timeMin`
  Start time of an unavailable time slot.

**timeMax**

End time of an unavailable time slot.

Signature

```java
public Datetime timeMax {get; set;}
```

Property Value

Type: `Datetime`

**timeMin**

Start time of an unavailable time slot.

Signature

```java
public Datetime timeMin {get; set;}
```

Property Value

Type: `Datetime`
Messaging Namespace

The **Messaging** namespace provides classes and methods for Salesforce outbound and inbound email functionality. The following are the classes in the **Messaging** namespace.

**IN THIS SECTION:**

- **AttachmentRetrievalOption Enum**
  Provides options for including attachment metadata only, attachment metadata and content, or excluding attachments.

- **Email Class (Base Email Methods)**
  Contains base email methods common to both single and mass email.

- **EmailFileAttachment Class**
  EmailFileAttachment is used in SingleEmailMessage to specify attachments passed in as part of the request, as opposed to existing documents in Salesforce.

- **InboundEmail Class**
  Represents an inbound email object.

- **InboundEmail.AuthenticationResult Class**
  Contains the authentication type and response for inbound emails.

- **InboundEmail.AuthenticationResultField Class**
  Contains field data from the authentication result response for inbound emails.

- **InboundEmail.BinaryAttachment Class**
  An InboundEmail object stores binary attachments in an InboundEmail.BinaryAttachment object.

- **InboundEmail.TextAttachment Class**
  An InboundEmail object stores text attachments in an InboundEmail.TextAttachment object.

- **InboundEmailResult Class**
  The InboundEmailResult object is used to return the result of the email service. If this object is null, the result is assumed to be successful.

- **InboundEnvelope Class**
  The InboundEnvelope object stores the envelope information associated with the inbound email, and has the following fields.

- **MassEmailMessage Class**
  Contains methods for sending mass email.

- **InboundEmail.Header Class**
  An InboundEmail object stores RFC 2822 email header information in an InboundEmail.Header object with the following properties.

- **PushNotification Class**
  PushNotification is used to configure push notifications and send them from an Apex trigger.

- **PushNotificationPayload Class**
  Contains methods to create the notification message payload for an Apple device.

- **CustomNotification Class**
  CustomNotification is used to create, configure, and send custom notifications from Apex code.

- **RenderEmailTemplateBodyResult Class**
  Contains the results for rendering email templates.
RenderEmailTemplateError Class
Represents an error that the RenderEmailTemplateBodyResult object can contain.

SendEmailError Class
Represents an error that the SendEmailResult object may contain.

SendEmailResult Class
Contains the result of sending an email message.

SingleEmailMessage Methods
Contains methods for sending single email messages.

AttachmentRetrievalOption Enum
Provides options for including attachment metadata only, attachment metadata and content, or excluding attachments.

Namespace
Messaging

Usage
Use these enum values with the renderStoredEmailTemplate(templateId, whold, whatld, attachmentRetrievalOption) method.

Enum Values
The following are the values of the Messaging.AttachmentRetrievalOption enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>METADATA_ONLY</td>
<td>Includes only the file name, content type, and the object ID in the fileAttachments property of Messaging.SingleEmailMessage.</td>
</tr>
<tr>
<td></td>
<td>Note: When the template is rendered from a Visualforce template (and not from a static file attached to the template), the object ID is not available.</td>
</tr>
<tr>
<td>METADATA_WITH_BODY</td>
<td>Includes the attachment content, in addition to the file name, content type, and the object ID in the fileAttachments property of Messaging.SingleEmailMessage.</td>
</tr>
<tr>
<td>NONE</td>
<td>Doesn’t include any attachments in Messaging.SingleEmailMessage.</td>
</tr>
</tbody>
</table>

Email Class (Base Email Methods)
Contains base email methods common to both single and mass email.

Namespace
Messaging
Usage

Note: If templates are not being used, all email content must be in plain text, HTML, or both. Visualforce email templates cannot be used for mass email.

Email Methods

The following are methods for Email. All are instance methods.

IN THIS SECTION:

- `setBccSender(bcc)`
- `setReplyTo(replyAddress)`
- `setTemplateID(templateId)`
- `setSaveAsActivity(saveAsActivity)`
- `setSenderDisplayName(displayName)`
- `setUseSignature(useSignature)`

`setBccSender(bcc)`

Indicates whether the email sender receives a copy of the email that is sent. For a mass mail, the sender is only copied on the first email sent.

Signature

```java
public Void setBccSender(Boolean bcc)
```

Parameters

- `bcc`
  
  Type: Boolean

Return Value

Type: Void
Usage

Note: If the BCC compliance option is set at the organization level, the user cannot add BCC addresses on standard messages. The following error code is returned: BCC_NOT_ALLOWED_IF_BCC_COMPLIANCE_ENABLED. Contact your Salesforce representative for information on BCC compliance.

setReplyTo(replyAddress)
Optional. The email address that receives the message when a recipient replies.

Signature
public Void setReplyTo(String replyAddress)

Parameters
replyAddress
Type: String

Return Value
Type: Void

setTemplateID(templateId)
The ID of the template to be merged to create this email. Specify a value for setTemplateId, setHtmlBody, or setPlainTextBody. Or, you can define both setHtmlBody and setPlainTextBody.

Signature
public Void setTemplateID(ID templateId)

Parameters
templateId
Type: ID

Return Value
Type: Void

Usage

Note: setHtmlBody and setPlainTextBody apply only to single email methods, not to mass email methods.

setSaveAsActivity(saveAsActivity)
Optional. The default value is true, meaning the email is saved as an activity. This argument only applies if the recipient list is based on targetObjectId or targetObjectIds. If HTML email tracking is enabled for the organization, you will be able to track open rates.
Signature

public Void setSaveAsActivity(Boolean saveAsActivity)

Parameters

saveAsActivity
Type: Boolean

Return Value
Type: Void

setSenderDisplayName(displayName)
Optional. The name that appears on the From line of the email. This cannot be set if the object associated with a setOrgWideEmailAddressId for a SingleEmailMessage has defined its DisplayName field.

Signature

public Void setSenderDisplayName(String displayName)

Parameters

displayName
Type: String

Return Value
Type: Void

setUseSignature(useSignature)
Indicates whether the email includes an email signature if the user has one configured. The default is true, meaning if the user has a signature it is included in the email unless you specify false.

Signature

public Void setUseSignature(Boolean useSignature)

Parameters

useSignature
Type: Boolean

Return Value
Type: Void
EmailFileAttachment Class

EmailFileAttachment is used in SingleEmailMessage to specify attachments passed in as part of the request, as opposed to existing documents in Salesforce.

Namespace

Messaging

IN THIS SECTION:

EmailFileAttachment Constructors

EmailFileAttachment Properties

EmailFileAttachment Constructors

The following are constructors for EmailFileAttachment.

IN THIS SECTION:

EmailFileAttachment()

EmailFileAttachment()

Signature

public EmailFileAttachment()

EmailFileAttachment Properties

The following are properties for EmailFileAttachment.

IN THIS SECTION:

body

contenttype

filename

id

inline

Specifies a Content-Disposition of inline (true) or attachment (false).
**body**
Gets or sets the attachment itself.

**Signature**
```java
public Blob body {get; set;}
```

**Property Value**
Type: Blob

**contenttype**
Gets or sets the attachment’s Content-Type.

**Signature**
```java
public String contenttype {get; set;}
```

**Property Value**
Type: String

**filename**
Gets or sets the name of the file to attach.

**Signature**
```java
public String filename {get; set;}
```

**Property Value**
Type: String

**id**
Read-Only. Gets the attachment ID.

**Signature**
```java
public Id id {get;}
```

**Property Value**
Type: Id

**inline**
Specifies a Content-Disposition of inline (`true`) or attachment (`false`).
Signature

public Boolean inline {get; set;}

Property Value

Type: Boolean

InboundEmail Class

Represents an inbound email object.

Namespace

Messaging

IN THIS SECTION:
   InboundEmail Constructors
   InboundEmail Properties

InboundEmail Constructors

The following are constructors for InboundEmail.

IN THIS SECTION:
   InboundEmail()

InboundEmail()

Creates a new instance of the Messaging.InboundEmail class.

Signature

public InboundEmail()

InboundEmail Properties

The following are properties for InboundEmail.

IN THIS SECTION:
   authenticationResults
   A list of authentication results received with the email, if any.
   binaryAttachments
   A list of binary attachments received with the email, if any.
ccAddresses
A list of carbon copy (CC) addresses, if any.

cFrom
The email address that appears in the From field.

cFromName
The name that appears in the From field, if any.

cHeaders
A list of the RFC 2822 headers in the email.

cHtmlBody
The HTML version of the email, if specified by the sender.

cHtmlBodyIsTruncated
Indicates whether the HTML body text is truncated (true) or not (false).

cInReplyTo
The In-Reply-To field of the incoming email. Identifies the email or emails to which this one is a reply (parent emails). Contains the parent email or emails' message-IDs.

cMessageId
The Message-ID—the incoming email's unique identifier.

cPlainTextBody
The plain text version of the email, if specified by the sender.

cPlainTextBodyIsTruncated
Indicates whether the plain body text is truncated (true) or not (false).

cReferences
The References field of the incoming email. Identifies an email thread. Contains a list of the parent emails' References and message IDs, and possibly the In-Reply-To fields.

cReplyTo
The email address that appears in the reply-to header.

cSubject
The subject line of the email, if any.

cTextAttachments
A list of text attachments received with the email, if any.

cToAddresses
The email address that appears in the To field.

authenticationResults
A list of authentication results received with the email, if any.

Signature

public InboundEmail.AuthenticationResult[] authenticationResults {get; set;}

2305
Property Value
Type: InboundEmail.AuthenticationResult[]

Usage
Examples of authentication results include dkim, dmarc, and spf.

binaryAttachments
A list of binary attachments received with the email, if any.

Signature
public InboundEmail.BinaryAttachment[] binaryAttachments {get; set;}

Property Value
Type: InboundEmail.BinaryAttachment[]

Usage
Examples of binary attachments include image, audio, application, and video files.

ccAddresses
A list of carbon copy (CC) addresses, if any.

Signature
public String[] ccAddresses {get; set;}

Property Value
Type: String[]

fromAddress
The email address that appears in the From field.

Signature
public String fromAddress {get; set;}

Property Value
Type: String

fromName
The name that appears in the From field, if any.
Signature
```
public String fromName {get; set;}
```

Property Value
Type: String

headers
A list of the RFC 2822 headers in the email.

Signature
```
public InboundEmail.Header[] headers {get; set;}
```

Property Value
Type: InboundEmail.Header[]

Usage
The list of the RFC 2822 headers includes:
- Received from
- Custom headers
- Message-ID
- Date

htmlBody
The HTML version of the email, if specified by the sender.

Signature
```
public String htmlBody {get; set;}
```

Property Value
Type: String

htmlBodyIsTruncated
Indicates whether the HTML body text is truncated (true) or not (false)

Signature
```
public Boolean htmlBodyIsTruncated {get; set;}
```
**Property Value**

Type: `Boolean`

**inReplyTo**

The In-Reply-To field of the incoming email. Identifies the email or emails to which this one is a reply (parent emails). Contains the parent email or emails’ message-IDs.

**Signature**

```java
public String inReplyTo {get; set;}
```

**Property Value**

Type: `String`

**messageId**

The Message-ID—the incoming email’s unique identifier.

**Signature**

```java
public String messageId {get; set;}
```

**Property Value**

Type: `String`

**plainTextBody**

The plain text version of the email, if specified by the sender.

**Signature**

```java
public String plainTextBody {get; set;}
```

**Property Value**

Type: `String`

**plainTextBodyIsTruncated**

Indicates whether the plain body text is truncated (`true`) or not (`false`).

**Signature**

```java
public Boolean plainTextBodyIsTruncated {get; set;}
```
Property Value
Type: Boolean

**references**
The References field of the incoming email. Identifies an email thread. Contains a list of the parent emails' References and message IDs, and possibly the In-Reply-To fields.

Signature
```
public String[] references {get; set;}
```

Property Value
Type: String[]

**replyTo**
The email address that appears in the reply-to header.

Signature
```
public String replyTo {get; set;}
```

Property Value
Type: String

Usage
If there is no reply-to header, this field is identical to the `fromAddress` field.

**subject**
The subject line of the email, if any.

Signature
```
public String subject {get; set;}
```

Property Value
Type: String

**textAttachments**
A list of text attachments received with the email, if any.
InboundEmail.TextAttachment[] textAttachments {get; set;}

Property Value
Type: InboundEmail.TextAttachment[]

Usage
The text attachments can be any of the following:
- Attachments with a Multipurpose Internet Mail Extension (MIME) type of text
- Attachments with a MIME type of application/octet-stream and a file name that ends with either a .vcf or .vcs extension. These are saved as text/x-vcard and text/calendar MIME types, respectively.

toAddresses
The email address that appears in the To field.

InboundEmail.AuthenticationResult Class
Contains the authentication type and response for inbound emails.

Namespace
Messaging

InboundEmail.AuthenticationResult Constructors
The following are constructors for InboundEmail.AuthenticationResult.

InboundEmail.AuthenticationResult Class
Apex Reference Guide  InboundEmail.AuthenticationResult Class

InboundEmail.AuthenticationResult() Creates a new instance of the Messaging.InboundEmail.AuthenticationResult class.
InboundEmail.AuthenticationResult()

Creates a new instance of the Messaging.InboundEmail.AuthenticationResult class.

Signature

public InboundEmail.AuthenticationResult()

InboundEmail.AuthenticationResult Properties

The following are properties for InboundEmail.AuthenticationResult.

IN THIS SECTION:

  authenticationResultFields
  Additional information in authentication result headers. Examples include: name: smtp.mailfrom and value: example.com.

  method
  The authentication method used for the security check. Possible values include dkim, dmarc, or spf.

  result
  The result of the authentication check. When the email service is configured to verify the legitimacy of the sending server before processing a message, possible values include pass or fail. Otherwise, the value returned is none.

authenticationResultFields

Additional information in authentication result headers. Examples include: name: smtp.mailfrom and value: example.com.

Signature

public InboundEmail.AuthenticationResultField[] authenticationResultFields {get; set;}

Property Value

Type: InboundEmail.AuthenticationResultField[]

method

The authentication method used for the security check. Possible values include dkim, dmarc, or spf.

Signature

public String method {get; set;}

Property Value

Type: String
The result of the authentication check. When the email service is configured to verify the legitimacy of the sending server before processing a message, possible values include pass or fail. Otherwise, the value returned is none.

public String result {get; set;}

Type: String

InboundEmail.AuthenticationResultField Class

Contains field data from the authentication result response for inbound emails.

Namespace

Messaging

InboundEmail.AuthenticationResultField Constructors

The following are constructors for InboundEmail.AuthenticationResultField.

InboundEmail.AuthenticationResultField()

Creates a new instance of the Messaging.InboundEmail.AuthenticationResultField class.

InboundEmail.AuthenticationResultField()
IN THIS SECTION:

**name**
The authentication result field name. For example: `smtp.mailfrom`.

**value**
The authentication result field value. For example: `example.com`.

**Signature**

```java
public String name {get; set;}
```

**Property Value**

Type: `String`

**value**
The authentication result field value. For example: `example.com`.

**Signature**

```java
public String value {get; set;}
```

**Property Value**

Type: `String`

**InboundEmail.BinaryAttachment Class**

An `InboundEmail` object stores binary attachments in an `InboundEmail.BinaryAttachment` object.

**Namespace**

`Messaging`

**Usage**

Examples of binary attachments include image, audio, application, and video files.
InboundEmail.BinaryAttachment Constructors

The following are constructors for InboundEmail.BinaryAttachment.

IN THIS SECTION:

- InboundEmail.BinaryAttachment()
  Creates a new instance of the Messaging.InboundEmail.BinaryAttachment class.

InboundEmail.BinaryAttachment()

Creates a new instance of the Messaging.InboundEmail.BinaryAttachment class.

Signature

```java
public InboundEmail.BinaryAttachment()
```

InboundEmail.BinaryAttachment Properties

The following are properties for InboundEmail.BinaryAttachment.

IN THIS SECTION:

- body
  The body of the attachment.
- fileName
  The name of the attached file.
- headers
  Any header values associated with the attachment. Examples of header names include Content-Type, Content-Transfer-Encoding, and Content-ID.
- mimeTypeSubType
  The primary and sub MIME-type.

body

The body of the attachment.

Signature

```java
public Blob body {get; set;}
```

Property Value

Type: Blob

fileName

The name of the attached file.
**Signature**

```java
public String fileName {get; set;}
```

**Property Value**

Type: `String`

**headers**

Any header values associated with the attachment. Examples of header names include `Content-Type`, `Content-Transfer-Encoding`, and `Content-ID`.

**Signature**

```java
public List<Messaging.InboundEmail.Header> headers {get; set;}
```

**Property Value**

Type: `List<Messaging.InboundEmail.Header>`

**mimeTypeSubType**

The primary and sub MIME-type.

**Signature**

```java
public String mimeTypeSubType {get; set;}
```

**Property Value**

Type: `String`

---

**InboundEmail.TextAttachment Class**

An InboundEmail object stores text attachments in an InboundEmail.TextAttachment object.

**Namespace**

`Messaging`

**Usage**

The text attachments can be any of the following:

- Attachments with a Multipurpose Internet Mail Extension (MIME) type of `text`
- Attachments with a MIME type of `application/octet-stream` and a file name that ends with either a `.vcf` or `.vcs` extension. These are saved as `text/x-vcard` and `text/calendar` MIME types, respectively.
IN THIS SECTION:
  InboundEmail.TextAttachment Constructors
  InboundEmail.TextAttachment Properties

InboundEmail.TextAttachment Constructors

The following are constructors for InboundEmail.TextAttachment.

IN THIS SECTION:
  InboundEmail.TextAttachment()
  Creates a new instance of the Messaging.InboundEmail.TextAttachment class.

InboundEmail.TextAttachment()

Creates a new instance of the Messaging.InboundEmail.TextAttachment class.

Signature

public InboundEmail.TextAttachment()

InboundEmail.TextAttachment Properties

The following are properties for InboundEmail.TextAttachment.

IN THIS SECTION:
  body
  The body of the attachment.
  bodyIsTruncated
  Indicates whether the attachment body text is truncated (true) or not (false).
  charset
  The original character set of the body field. The body is re-encoded as UTF-8 as input to the Apex method.
  fileName
  The name of the attached file.
  headers
  Any header values associated with the attachment. Examples of header names include Content-Type, Content-Transfer-Encoding, and Content-ID.
  mimeTypeSubType
  The primary and sub MIME-type.

body

The body of the attachment.
Signature

public String body {get; set;}

Property Value
Type: String

**bodyIsTruncated**
Indicates whether the attachment body text is truncated (**true**) or not (**false**).

Signature

public Boolean bodyIsTruncated {get; set;}

Property Value
Type: Boolean

**charset**
The original character set of the body field. The body is re-encoded as UTF-8 as input to the Apex method.

Signature

public String charset {get; set;}

Property Value
Type: String

**fileName**
The name of the attached file.

Signature

public String fileName {get; set;}

Property Value
Type: String

**headers**
Any header values associated with the attachment. Examples of header names include **Content-Type**, **Content-Transfer-Encoding**, and **Content-ID**.
Signature
public List<Messaging.InboundEmail.Header> headers {get; set;}

Property Value
Type: List<Messaging.InboundEmail.Header>

mimeTypeSubType
The primary and sub MIME-type.

Signature
public String mimeTypeSubType {get; set;}

Property Value
Type: String

InboundEmailResult Class
The InboundEmailResult object is used to return the result of the email service. If this object is null, the result is assumed to be successful.

Namespace
Messaging

InboundEmailResult Properties
The following are properties for InboundEmailResult.

IN THIS SECTION:
message
A message that Salesforce returns in the body of a reply email. This field can be populated with text irrespective of the value returned by the Success field.

success
A value that indicates whether the email was successfully processed.

message
A message that Salesforce returns in the body of a reply email. This field can be populated with text irrespective of the value returned by the Success field.

Signature
public String message {get; set;}

2318
Property Value
Type: String

**success**
A value that indicates whether the email was successfully processed.

Signature
```csharp
public Boolean success {get; set;}
```

Property Value
Type: Boolean

Usage
If false, Salesforce rejects the inbound email and sends a reply email to the original sender containing the message specified in the Message field.

**InboundEnvelope Class**
The InboundEnvelope object stores the envelope information associated with the inbound email, and has the following fields.

**Namespace**
Messaging

**InboundEnvelope Properties**
The following are properties for InboundEnvelope.

IN THIS SECTION:
- **fromAddress**
The name that appears in the From field of the envelope, if any.
- **toAddress**
The name that appears in the To field of the envelope, if any.

**fromAddress**
The name that appears in the From field of the envelope, if any.

Signature
```csharp
public String fromAddress {get; set;}
```
Property Value
Type: String

toAddress
The name that appears in the To field of the envelope, if any.

Signature
public String toAddress {get; set;}

Property Value
Type: String

MassEmailMessage Class
Contains methods for sending mass email.

Namespace
Messaging

Usage
MassEmailMessage extends Email and inherits all of its methods. All base email (Email class) methods are also available to the MassEmailMessage objects.

IN THIS SECTION:
MassEmailMessage Constructors
MassEmailMessage Methods

SEE ALSO:
Email Class (Base Email Methods)

MassEmailMessage Constructors
The following are constructors for MassEmailMessage.

IN THIS SECTION:
MassEmailMessage()
MassEmailMessage()

MassEmailMessage()
Creates a new instance of the Messaging.MassEmailMessage class.
Signature
public MassEmailMessage()

MassEmailMessage Methods
The following are methods for MassEmailMessage. All are instance methods. All base email (Email class) methods are also available to the MassEmailMessage objects. These methods are described in Email Class (Base Email Methods).

IN THIS SECTION:

setDescription(String)
The description of the email.

setTargetObjectIds(ID[])
A list of IDs of the contacts, leads, or users to which the email will be sent. The IDs you specify set the context and ensure that merge fields in the template contain the correct data. The objects must be of the same type (all contacts, all leads, or all users).

setWhatIds(ID[])
Optional. If you specify a list of contacts for the targetObjectIds field, you can specify a list of whatIds as well. This helps to further ensure that merge fields in the template contain the correct data.

setDescription(String)
The description of the email.

Signature
public Void setDescription(String description)

Parameters
description
Type: String

Return Value
Type: Void

setTargetObjectIds(ID[])
A list of IDs of the contacts, leads, or users to which the email will be sent. The IDs you specify set the context and ensure that merge fields in the template contain the correct data. The objects must be of the same type (all contacts, all leads, or all users).

Signature
public Void setTargetObjectIds(ID[] targetObjectIds)

Parameters
targetObjectIds
Type: ID[]
Return Value
Type: Void

Usage
You can list multiple IDs per email. If you specify a value for the targetObjectIds field, optionally specify a whatId as well to set the email context to a user, contact, or lead. This ensures that merge fields in the template contain the correct data. Each ID counts against the sending organization’s daily mass email limit.
Do not specify the IDs of records that have the Email Opt Out option selected.
All emails must have a recipient value in at least one of the following fields:
- toAddresses
- ccAddresses
- bccAddresses
- targetObjectId

setWhatIds (whatIds)
Optional. If you specify a list of contacts for the targetObjectIds field, you can specify a list of whatIds as well. This helps to further ensure that merge fields in the template contain the correct data.

Signature
public Void setWhatIds(ID[] whatIds)

Parameters
whatIds
Type: ID[]

Return Value
Type: Void

Usage
The values must be one of the following types:
- Contract
- Case
- Opportunity
- Product

Note: If you specify whatIds, specify one for each targetObjectId; otherwise, you will receive an INVALID_ID_FIELD error.

InboundEmail.Header Class
An InboundEmail object stores RFC 2822 email header information in an InboundEmail.Header object with the following properties.
Namespace
Messaging

InboundEmail.Header Properties
The following are properties for InboundEmail.Header.

IN THIS SECTION:

name
The name of the header parameter, such as Date or Message-ID.

type
The type of the header.

value
The value of the header.

signature
public String name {get; set;}

property value
Type: String

value
The value of the header.

signature
public String value {get; set;}

property value
Type: String

PushNotification Class

PushNotification is used to configure push notifications and send them from an Apex trigger.

Namespace
Messaging
Example

This sample Apex trigger sends push notifications to the connected app named Test_App, which corresponds to a mobile app on iOS mobile clients. The trigger fires after cases have been updated and sends the push notification to two users: the case owner and the user who last modified the case.

trigger caseAlert on Case (after update) {
    for(Case cs : Trigger.New) {
        // Instantiating a notification

        // Assembling the necessary payload parameters for Apple.
        // Apple params are:
        // ({alert text},{alert sound},{badge count},
        // {free-form data})
        // This example doesn't use badge count or free-form data.
        // The number of notifications that haven't been acted
        // upon by the intended recipient is best calculated
        // at the time of the push. This timing helps
        // ensure accuracy across multiple target devices.
        Map<String, Object> payload = Messaging.PushNotificationPayload.apple('Case ' + cs.CaseNumber + ' status changed to: ' + cs.Status, '', null, null);

        // Adding the assembled payload to the notification
        msg.setPayload(payload);

        // Getting recipient users
        String userId1 = cs.OwnerId;
        String userId2 = cs.LastModifiedById;

        // Adding recipient users to list
        Set<String> users = new Set<String>();
        users.add(userId1);
        users.add(userId2);

        // Sending the notification to the specified app and users.
        // Here we specify the API name of the connected app.
        msg.send('Test_App', users);
    }
}
PushNotification Constructors

The following are the constructors for PushNotification.

IN THIS SECTION:

PushNotification()
Creates a new instance of the Messaging.PushNotification class.

PushNotification(payload)
Creates a new instance of the Messaging.PushNotification class using the specified payload parameters as key-value pairs. When you use this constructor, you don’t need to call setPayload to set the payload.

PushNotification()

Creates a new instance of the Messaging.PushNotification class.

Signature

public PushNotification()

PushNotification (payload)

Creates a new instance of the Messaging.PushNotification class using the specified payload parameters as key-value pairs. When you use this constructor, you don’t need to call setPayload to set the payload.

Signature

public PushNotification (Map<String, Object> payload)

Parameters

payload
Type: Map<String, Object>
The payload, expressed as a map of key-value pairs.

PushNotification Methods

The following are the methods for PushNotification. All are global methods.

IN THIS SECTION:

send(application, users)
Sends a push notification message to the specified users.

setPayload(payload)
Sets the payload of the push notification message.

setTtl(ttl)
Reserved for future use.
**send(application, users)**
Sends a push notification message to the specified users.

**Signature**
```
public void send(String application, Set<String> users)
```

**Parameters**
- **application**
  - Type: String
  - The connected app API name. This corresponds to the mobile client app the notification should be sent to.
- **users**
  - Type: Set
  - A set of user IDs that correspond to the users the notification should be sent to.

**Example**
See the Push Notification Example.

**setPayload(payload)**
Sets the payload of the push notification message.

**Signature**
```
public void setPayload(Map<String, Object> payload)
```

**Parameters**
- **payload**
  - Type: Map<String, Object>
  - The payload, expressed as a map of key-value pairs.
  - Payload parameters can be different for each mobile OS vendor. For more information on Apple's payload parameters, search for "Apple Push Notification Service" at https://developer.apple.com/library/mac/documentation/. To create the payload for an Apple device, see the PushNotificationPayload Class.

**Example**
See the Push Notification Example.

**setTtl(ttl)**
Reserved for future use.

**Signature**
```
public void setTtl(Integer ttl)
```
Parameters

ttl
Type: Integer
Reserved for future use.

**PushNotificationPayload Class**
Contains methods to create the notification message payload for an Apple device.

**Namespace**
Messaging

**Usage**
Apple has specific requirements for the notification payload, and this class has helper methods to create the payload. For more information on Apple’s payload parameters, search for “Apple Push Notification Service” at [https://developer.apple.com/library/mac/documentation/](https://developer.apple.com/library/mac/documentation/).

**Example**
See the Push Notification Example.

IN THIS SECTION:
PushNotificationPayload Methods

**PushNotificationPayload Methods**
The following are the methods for PushNotificationPayload. All are global static methods.

IN THIS SECTION:

apple(alert, sound, badgeCount, userData)
Helper method that creates a valid Apple payload from the specified arguments.

apple(alertBody, actionLocKey, locKey, locArgs, launchImage, sound, badgeCount, userData)
Helper method that creates a valid Apple payload from the specified arguments.

**apple(alert, sound, badgeCount, userData)**
Helper method that creates a valid Apple payload from the specified arguments.

**Signature**

```java
public static Map<String,Object> apple(String alert, String sound, Integer badgeCount, Map<String,Object> userData)
```
Parameters

alert
  Type: String
  Notification message to be sent to the mobile client.

sound
  Type: String
  Name of a sound file to be played as an alert. This sound file should be in the mobile application bundle.

badgeCount
  Type: Integer
  Number to display as the badge of the application icon.

userData
  Type: Map<String, Object>
  Map of key-value pairs that contains any additional data used to provide context for the notification. For example, it can contain IDs of the records that caused the notification to be sent. The mobile client app can use these IDs to display these records.

Return Value

Type: Map<String, Object>

Returns a formatted payload that includes all of the specified arguments.

Usage

To generate a valid payload, you must provide a value for at least one of the following parameters: alert, sound, badgeCount.

Example

See the Push Notification Example.

apple(alertBody, actionLocKey, locKey, locArgs, launchImage, sound, badgeCount, userData)

Helper method that creates a valid Apple payload from the specified arguments.

Signature

public static Map<String, Object> apple(String alertBody, String actionLocKey, String locKey, String[] locArgs, String launchImage, String sound, Integer badgeCount, Map<String, Object> userData)

Parameters

alertBody
  Type: String
  Text of the alert message.

actionLocKey
  Type: String
If a value is specified for the `actionLocKey` argument, an alert with two buttons is displayed. The value is a key to get a localized string in a `Localizable.strings` file to use for the right button’s title.

`locKey`
- **Type**: `String`
- Key to an alert-message string in a `Localizable.strings` file for the current localization.

`locArgs`
- **Type**: `List<String>`
- Variable string values to appear in place of the format specifiers in `locKey`.

`launchImage`
- **Type**: `String`
- File name of an image file in the application bundle.

`sound`
- **Type**: `String`
- Name of a sound file to be played as an alert. This sound file should be in the mobile application bundle.

`badgeCount`
- **Type**: `Integer`
- Number to display as the badge of the application icon.

`userData`
- **Type**: `Map<String, Object>`
- Map of key-value pairs that contains any additional data used to provide context for the notification. For example, it can contain IDs of the records that caused the notification to be sent. The mobile client app can use these IDs to display these records.

**Return Value**
- **Type**: `Map<String, Object>`
- Returns a formatted payload that includes all of the specified arguments.

**Usage**
To generate a valid payload, you must provide a value for at least one of the following parameters: `alert`, `sound`, `badgeCount`.

**CustomNotification Class**

*CustomNotification* is used to create, configure, and send custom notifications from Apex code.

**Namespace**

*.Messaging*

**Usage**

*CustomNotification* allows two approaches to creating and configuring a custom notification.

- Create an instance with the default constructor, and then set notification attributes using the various setter methods.
- Create an instance and configure notification parameters at the same time using the parameterized constructor.
Once the custom notification is configured, call send() to send the notification.

**Notification Target**

The notification target is used by the receiving client application to navigate to an appropriate record or page when a user responds to a notification. For example, when a user is notified that a record was updated, responding to the notification can open the relevant record.

You must specify a target for a notification. The target can be specified using either the targetID or the targetPageRef attribute. Neither attribute is required, but if both are omitted, send() throws an exception. If there's no natural target for a notification, set the targetID to a dummy value, such as 000000000000000AAA. A dummy value prevents the exception, and also prevents automatic navigation when responding to the notification in the client app.

You can set both targetID and targetPageRef in the same notification. The client app that receives the notification determines which target, if any, to use when responding to the notification.

**Important:** Before Winter '21 you could set only a target record (targetID) for a notification. Most client applications expect to find a targetID in the notification payload. If you can't update a client app to handle notifications that include only a targetPageRef, set the targetID to a dummy value.

**Execution Context and Notification Permissions**

By default Apex code executes in system mode, and doesn't require user permissions to send notifications with CustomNotification. However, if your Apex code runs in a user context—for example, by executing anonymous Apex in the Developer Console—the Send Custom Notifications user permission is checked, and send() fails if you don't have the required permission.

**Example**

This example Apex class provides a static method for sending a custom notification to a recipient list. Call this method from a trigger, flow, or wherever you want to send a custom notification from Apex.

```apex
public without sharing class CustomNotificationFromApex {
    public static void notifyUsers(Set<String> recipientsIds, String targetId) {
        // Get the Id for our custom notification type
        CustomNotificationType notificationType =
            [SELECT Id, DeveloperName
             FROM CustomNotificationType
             WHERE DeveloperName='Custom_Notification'];

        // Create a new custom notification
        Messaging.CustomNotification notification = new Messaging.CustomNotification();

        // Set the contents for the notification
        notification.setTitle('Apex Custom Notification');
        notification.setBody('The notifications are coming from INSIDE the Apex!');

        // Set the notification type and target
        notification.setNotificationTypeId(notificationType.Id);
        notification.setTargetId(targetId);

        // Actually send the notification
        try {
            notification.send(recipientsIds);
        }
    }
}
```
```java
    catch (Exception e) {
        System.debug('Problem sending notification: ' + e.getMessage());
    }
}

Note: This example uses a hard-coded string, Custom_Notification, as the DeveloperName (also known as the API Name) of a Custom Notification Type. Use your custom notification types in your own code.

    CustomNotification.send() can throw an exception, which is handled minimally in this example. Add more substantial error handling to code you plan to use in production.

IN THIS SECTION:

    CustomNotification Constructors
    CustomNotification Methods

SEE ALSO:

    Salesforce Help: Send Custom Notifications
    Actions Developer Guide: Custom Notification Actions
    Metadata API Developer Guide: CustomNotificationType

CustomNotification Constructors

The following are constructors for CustomNotification.

IN THIS SECTION:

    CustomNotification()
    Creates a new instance of the Messaging.CustomNotification class.

    CustomNotification(typeId, sender, title, body, targetId, targetPageRef)
    Creates an instance of the Messaging.CustomNotification class using the specified parameters. When you use this constructor, you don’t need to call the various setter methods to define the custom notification attributes.

CustomNotification()

Creates a new instance of the Messaging.CustomNotification class.

Signature

public CustomNotification()

CustomNotification(typeId, sender, title, body, targetId, targetPageRef)

Creates an instance of the Messaging.CustomNotification class using the specified parameters. When you use this constructor, you don’t need to call the various setter methods to define the custom notification attributes.
Signature

```java
public CustomNotification(String typeId, String sender, String title, String body,
String targetId, String targetPageRef)
```

Parameters

typeId
Type: String
The ID of the Custom Notification Type being used for the notification.

sender
Type: String
The User ID of the sender of the notification.

title
Type: String
The title of the notification. Maximum characters: 250.

body
Type: String
The body of the notification. Maximum characters: 750.

targetId
Type: String
The Record ID for the target record of the notification.

You must specify either a `targetID` or a `targetPageRef`. See Custom Notification Usage.

targetPageRef
Type: String
The `PageReference` for the navigation target of the notification. To see how to specify the target using JSON, see pageReference Types.

You must specify either a `targetID` or a `targetPageRef`. See Custom Notification Usage.

Usage

A client may see a truncated notification title or body depending on the delivery channel or app, and how the Connect API notification parameters are configured. For more information on the `trimMessages` query parameter, see Notification.

CustomNotification Methods

The following are methods for CustomNotification.

IN THIS SECTION:

- send(users)
  Sends a custom notification to the specified users.
- setNotificationTypeId(id)
  Sets the type of the custom notification.
setTitle(title)
Sets the title of the custom notification.

setBody(body)
Sets the body of the custom notification.

setSenderId(id)
Sets the sender of the custom notification.

setTargetId(targetId)
Sets the target record of the custom notification.

setTargetPageRef(pageRef)
Sets the target page of the custom notification.

send(users)
Sends a custom notification to the specified users.

Signature
public void send(Set<String> users)

Parameters
users
Type: Set<String>
Required. A set of recipient IDs. Each recipient ID corresponds to a recipient or recipient type that the notification should be sent to. Valid recipient or recipient type values are:

- **UserId** — The notification is sent to this user, if this user is active.
- **AccountId** — The notification is sent to all active users who are members of this account’s Account Team.

⚠️ **Note:** This recipient type is valid if account teams are enabled for your org.

- **OpportunityId** — The notification is sent to all active users who are members of this opportunity’s Opportunity Team.

⚠️ **Note:** This recipient type is valid if team selling is enabled for your org.

- **GroupId** — The notification is sent to all active users who are members of this group.
- **QueueId** — The notification is sent to all active users who are members of this queue.

Values can be combined in a set, up to the maximum of 500 values.

Return Value
Type: void

Example
See the Custom Notification Example.
**setNotificationTypeId(id)**
Sets the type of the custom notification.

**Signature**
```java
public void setNotificationTypeId(String id)
```

**Parameters**
- **id**
  - Type: String
  - The ID of the Custom Notification Type being used for the notification.
  - A notification type is required to send a custom notification. See Custom Notification Usage.

**Return Value**
Type: void

**Example**
See the Custom Notification Example.

**setTitle(title)**
Sets the title of the custom notification.

**Signature**
```java
public void setTitle(String title)
```

**Parameters**
- **title**
  - Type: String
  - The title of the notification, as it will be seen by recipients. Maximum characters: 250.
  - A title is required to send a custom notification. See Custom Notification Usage.

**Return Value**
Type: void

**Example**
See the Custom Notification Example.

**setBody (body)**
Sets the body of the custom notification.
Signature
public void setBody(String body)

Parameters
body
   Type: String
   The body of the notification, as it will be seen by recipients. Maximum characters: 750.
   A body is required to send a custom notification. See Custom Notification Usage.

Return Value
Type: void

Example
See the Custom Notification Example.

setSenderId(id)
Sets the sender of the custom notification.

Signature
public void setSenderId(String id)

Parameters
id
   Type: String
   The User ID of the sender of the notification.
   Setting a sender is optional. See Custom Notification Usage.

Return Value
Type: void

Example
See the Custom Notification Example.

setTargetId(targetId)
Sets the target record of the custom notification.

Signature
public void setTargetId(String targetId)
Parameters

`targetId`
Type: `String`
The Record ID for the target record of the notification.
Either a `targetID` or a `targetPageRef` is required to send a custom notification. See Custom Notification Usage.

Return Value
Type: `void`

Example
See the Custom Notification Example.

```java
setTargetPageRef(pageRef)
```
Sets the target page of the custom notification.

Signature
```java
public void setTargetPageRef(String pageRef)
```

Parameters

`pageRef`
Type: `String`
The PageReference for the navigation target of the notification.
Either a `targetID` or a `targetPageRef` is required to send a custom notification. See Custom Notification Usage.

Return Value
Type: `void`

Example
See the Custom Notification Example.

`RenderEmailTemplateBodyResult Class`
Contains the results for rendering email templates.

Namespace
`Messaging`

IN THIS SECTION:
- `RenderEmailTemplateBodyResult Methods`
RenderEmailTemplateBodyResult Methods

The following are methods for RenderEmailTemplateBodyResult.

IN THIS SECTION:

getErrors()
If an error occurred during the renderEmailTemplate method, a RenderEmailTemplateError object is returned.

getMergedBody()
Returns the rendered body text with merge field references replaced with the corresponding record data.

getSuccess()
Indicates whether the operation was successful.

getErrors()
If an error occurred during the renderEmailTemplate method, a RenderEmailTemplateError object is returned.

Signature

public List<Messaging.RenderEmailTemplateError> getErrors()

Return Value

Type: List<Messaging.RenderEmailTemplateError>

getMergedBody()
Returns the rendered body text with merge field references replaced with the corresponding record data.

Signature

public String getMergedBody()

Return Value

Type: String

getSuccess()
Indicates whether the operation was successful.

Signature

public Boolean getSuccess()

Return Value

Type: Boolean
RenderEmailTemplateError Class

Represents an error that the RenderEmailTemplateBodyResult object can contain.

Namespace

Messaging

IN THIS SECTION:

RenderEmailTemplateError Methods

RenderEmailTemplateError Methods

The following are methods for RenderEmailTemplateError.

IN THIS SECTION:

getFieldName()
Returns the name of the merge field in the error.

getMessage()
Returns a message describing the error.

getOffset()
Returns the offset within the supplied body text where the error was discovered. If the offset cannot be determined, -1 is returned.

getStatusCode()
Returns a Salesforce API status code.

getFieldName()

Returns the name of the merge field in the error.

Signature

public String getFieldName()

Return Value

Type: String

getMessage()

Returns a message describing the error.

Signature

public String getMessage()
Return Value
Type: String

getOffset()
Returns the offset within the supplied body text where the error was discovered. If the offset cannot be determined, -1 is returned.

Signature
public Integer getOffset()

Return Value
Type: Integer

getStatusCode()
Returns a Salesforce API status code.

Signature
public System.StatusCode getStatusCode()

Return Value
Type: System.StatusCode

SendEmailError Class
Represents an error that the SendEmailResult object may contain.

Namespace
Messaging

SendEmailError Methods
The following are methods for SendEmailError. All are instance methods.

IN THIS SECTION:
getFields()
A list of one or more field names. Identifies which fields in the object, if any, affected the error condition.

getMessage()
The text of the error message.

getStatusCode()
Returns a code that characterizes the error.
getTargetObjectId()
The ID of the target record for which the error occurred.

getFields()
A list of one or more field names. Identifies which fields in the object, if any, affected the error condition.

**Signature**
public String[] getFields()

**Return Value**
Type: String[]

getMessage()
The text of the error message.

**Signature**
public String getMessage()

**Return Value**
Type: String

getStatusCode()
Returns a code that characterizes the error.

**Signature**
public System.StatusCode getStatusCode()

**Return Value**
Type: System.StatusCode

**Usage**
The full list of status codes is available in the WSDL file for your organization. For more information about accessing the WSDL file for your organization, see Downloading Salesforce WSDLs and Client Authentication Certificates in the Salesforce online help.

getTargetObjectId()
The ID of the target record for which the error occurred.

**Signature**
public String getTargetObjectId()
SendEmailResult Class

Contains the result of sending an email message.

Namespace

Messaging

SendEmailResult Methods

The following are methods for SendEmailResult. All are instance methods.

IN THIS SECTION:

getErrors()
If an error occurred during the sendEmail method, a SendEmailError object is returned.

isSuccess()
Indicates whether the email was successfully submitted for delivery (true) or not (false). Even if isSuccess is true, it does not mean the intended recipients received the email, as there could have been a problem with the email address or it could have bounced or been blocked by a spam blocker.

getErrors()
If an error occurred during the sendEmail method, a SendEmailError object is returned.

Signature

public SendEmailError[] getErrors()

Return Value

Type: Messaging.SendEmailError[]

isSuccess()
Indicates whether the email was successfully submitted for delivery (true) or not (false). Even if isSuccess is true, it does not mean the intended recipients received the email, as there could have been a problem with the email address or it could have bounced or been blocked by a spam blocker.

Signature

public Boolean isSuccess()

Return Value

Type: Boolean
**SingleEmailMessage Methods**

Contains methods for sending single email messages.

**Namespace**

**Messaging**

**Usage**

SingleEmailMessage extends Email and inherits all of its methods. All base email (Email class) methods are also available to the SingleEmailMessage objects. Emails sent via SingleEmailMessage count against the sending organization's daily single email limit.

Email properties are readable and writable. Each property has corresponding setter and getter methods. For example, the `toAddresses()` property is equivalent to the `setToAddresses()` and `getToAddresses()` methods. Only the setter methods are documented. However, the `getTemplateName()` method doesn't have an equivalent setter method; use `setTemplateId()` to specify a template name.

**IN THIS SECTION:**

SingleEmailMessage Constructors

SingleEmailMessage Methods

**SEE ALSO:**

Email Class (Base Email Methods)

**SingleEmailMessage Constructors**

The following are constructors for `SingleEmailMessage`.

**IN THIS SECTION:**

`SingleEmailMessage()`

Creates a new instance of the `Messaging.SingleEmailMessage` class.

**SingleEmailMessage()**

Creates a new instance of the `Messaging.SingleEmailMessage` class.

**Signature**

```java
public SingleEmailMessage()
```

**SingleEmailMessage Methods**

The following are methods for `SingleEmailMessage`. All are instance methods. All base email (Email class) methods are also available to the `SingleEmailMessage` objects. These methods are described in Email Class (Base Email Methods).
IN THIS SECTION:

getTemplateName()
The name of the template used to create the email.

setBccAddresses(bccAddresses)
Optional. A list of blind carbon copy (BCC) addresses or object IDs of the contacts, leads, and users you’re sending the email to. The maximum size for this field is 4,000 bytes. The maximum total of toAddresses, ccAddresses, and bccAddresses per email is 150. All recipients in these three fields count against the limit for email sent using Apex or the API.

setCcAddresses(ccAddresses)
Optional. A list of carbon copy (CC) addresses or object IDs of the contacts, leads, and users you’re sending the email to. The maximum size for this field is 4,000 bytes. The maximum total of toAddresses, ccAddresses, and bccAddresses per email is 150. All recipients in these three fields count against the limit for email sent using Apex or the API.

setCharset(characterSet)
Optional. The character set for the email. If this value is null, the user’s default value is used.

setDocumentAttachments(documentIds)
(Deprecated. Use setEntityAttachments() instead.) Optional. A list containing the ID of each document object you want to attach to the email.

setEntityAttachments(ids)
Optional. Array of IDs of Document, ContentVersion, or Attachment items to attach to the email.

setFileAttachments(fileNames)
Optional. A list containing the file names of the binary and text files you want to attach to the email.

setHtmlBody(htmlBody)
Optional. The HTML version of the email, specified by the sender. The value is encoded according to the specification associated with the organization. Specify a value for setTemplateId, setHtmlBody, or setPlainTextBody. Or, you can define both setHtmlBody and setPlainTextBody.

setInReplyTo(parentMessageIds)
Sets the optional In-Reply-To field of the outgoing email. This field identifies the email or emails to which this email is a reply (parent emails).

setOptOutPolicy(emailOptOutPolicy)
Optional. If you added recipients by ID instead of email address and the Email Opt-Out option is set, this method determines the behavior of the sendEmail() call. If you add recipients by their email addresses, the opt-out settings for those recipients aren’t checked and those recipients always receive the email.

setPlainTextBody(plainTextBody)
Optional. The text version of the email, specified by the sender. Specify a value for setTemplateId, setHtmlBody, or setPlainTextBody. Or, you can define both setHtmlBody and setPlainTextBody.

setOrgWideEmailAddressId(emailAddressId)
Optional. The ID of the organization-wide email address associated with the outgoing email. If you’re using Apex to send emails from the guest user, set the sender to the verified org-wide email address or the emails are blocked. The object’s DisplayName field cannot be set if the setSenderDisplayName field is already set.

setReferences(references)
Optional. The References field of the outgoing email. Identifies an email thread. Contains the parent emails’ References and message IDs, and possibly the In-Reply-To fields.

setSubject(subject)
Optional. The email subject line. If you are using an email template, the subject line of the template overrides this value.
**setTargetObjectId(targetObjectId)**
Required if using a template, optional otherwise. The ID of the contact, lead, or user to which the email will be sent. The ID you specify sets the context and ensures that merge fields in the template contain the correct data.

**setTemplateId(templateId)**
Required if using a template, optional otherwise. The ID of the template used to create the email.

**setToAddresses(toAddresses)**
Optional. A list of email addresses or object IDs of the contacts, leads, and users you’re sending the email to. The maximum size for this field is 4,000 bytes. The maximum total of toAddresses, ccAddresses, and bccAddresses per email is 150. All recipients in these three fields count against the limit for email sent using Apex or the API.

**setTreatBodiesAsTemplate(treatAsTemplate)**
Optional. If set to true, the subject, plain text, and HTML text bodies of the email are treated as template data. The merge fields are resolved using the renderEmailTemplate method. Default is false.

**setTreatObjectIdAsRecipient(treatAsRecipient)**
Optional. If set to true, the targetObjectId (a contact, lead, or user) is the recipient of the email. If set to false, the targetObjectId is supplied as the WhoId field for template rendering but isn’t a recipient of the email. The default is true.

**setWhatId(whatId)**
If you specify a contact for the targetObjectId field, you can specify an optional whatId as well. This helps to further ensure that merge fields in the template contain the correct data.

**getTemplateName()**
The name of the template used to create the email.

**Signature**
```java
public STRING getTemplateName()
```

**Return Value**
Type: String

**Usage**
There is no equivalent setter method for getTemplateName(). If the email didn’t use a template, getTemplateName() returns nothing. If you use setTemplateId(), and then call getTemplateName(), the template name associated to the template ID is returned.

**setBccAddresses(bccAddresses)**
Optional. A list of blind carbon copy (BCC) addresses or object IDs of the contacts, leads, and users you’re sending the email to. The maximum size for this field is 4,000 bytes. The maximum total of toAddresses, ccAddresses, and bccAddresses per email is 150. All recipients in these three fields count against the limit for email sent using Apex or the API.

**Signature**
```java
public Void setBccAddresses(String[] bccAddresses)
```
Parameters

`bccAddresses`
Type: `String[]`

Return Value
Type: Void

Usage
All emails must have a recipient value in at least one of the following fields:

- `toAddresses`
- `ccAddresses`
- `bccAddresses`
- `targetObjectId`

If the BCC compliance option is set at the organization level, the user cannot add BCC addresses on standard messages. The following error code is returned: `BCC_NOT_ALLOWED_IF_BCC_COMPLIANCE_ENABLED`. Contact your Salesforce representative for information on BCC compliance.

`setCcAddresses(ccAddresses)`
Optional. A list of carbon copy (CC) addresses or object IDs of the contacts, leads, and users you’re sending the email to. The maximum size for this field is 4,000 bytes. The maximum total of `toAddresses`, `ccAddresses`, and `bccAddresses` per email is 150. All recipients in these three fields count against the limit for email sent using Apex or the API.

Signature

```
public Void setCcAddresses(String[] ccAddresses)
```

Parameters

`ccAddresses`
Type: `String[]`

Return Value
Type: Void

Usage
All emails must have a recipient value in at least one of the following fields:

- `toAddresses`
- `ccAddresses`
- `bccAddresses`
- `targetObjectId`
**setCharset(characterSet)**
Optional. The character set for the email. If this value is null, the user's default value is used.

**Signature**
```
public Void setCharset(String characterSet)
```

**Parameters**
`characterSet`
Type: `String`

**Return Value**
Type: `Void`

**setDocumentAttachments(documentIds)**
(Optional. Use `setEntityAttachments()` instead.) Optional. A list containing the ID of each document object you want to attach to the email.

**Signature**
```
public Void setDocumentAttachments(ID[] documentIds)
```

**Parameters**
`documentIds`
Type: `ID[]`

**Return Value**
Type: `Void`

**Usage**
You can attach multiple documents as long as the total size of all attachments does not exceed 10 MB.

**setEntityAttachments(ids)**
Optional. Array of IDs of `Document`, `ContentVersion`, or `Attachment` items to attach to the email.

**Signature**
```
public void setEntityAttachments(List<String> ids)
```

**Parameters**
`ids`
Type: `List<String>`
Return Value
Type: void

**setFileAttachments***(fileNames)*
Optional. A list containing the file names of the binary and text files you want to attach to the email.

**Signature**
```
public Void setFileAttachments(EmailFileAttachment[] fileNames)
```

**Parameters**
```
fileNames
  Type: Messaging.EmailFileAttachment[]
```

**Return Value**
Type: Void

**Usage**
You can attach multiple files as long as the total size of all attachments does not exceed 10 MB.

**setHtmlBody***(htmlBody)*
Optional. The HTML version of the email, specified by the sender. The value is encoded according to the specification associated with the organization. Specify a value for `setTemplateId`, `setHtmlBody`, or `setPlainTextBody`. Or, you can define both `setHtmlBody` and `setPlainTextBody`.

**Signature**
```
public Void setHtmlBody(String htmlBody)
```

**Parameters**
```
htmlBody
  Type: String
```

**Return Value**
Type: Void

**setInReplyTo***(parentMessageIds)*
Sets the optional In-Reply-To field of the outgoing email. This field identifies the email or emails to which this email is a reply (parent emails).
**Signature**

```java
public Void setInReplyTo(String parentMessageIds)
```

**Parameters**

`parentMessageIds`

Type: `String`  
Contains one or more parent email message IDs.

**Return Value**

Type: `Void`

**setOptOutPolicy(emailOptOutPolicy)**

Optional. If you added recipients by ID instead of email address and the Email Opt Out option is set, this method determines the behavior of the `sendEmail()` call. If you add recipients by their email addresses, the opt-out settings for those recipients aren’t checked and those recipients always receive the email.

**Signature**

```java
public void setOptOutPolicy(String emailOptOutPolicy)
```

**Parameters**

`emailOptOutPolicy`

Type: `String`  
Possible values of the `emailOptOutPolicy` parameter are:

- **SEND** (default)—The email is sent to all recipients. The recipients’ Email Opt Out setting is ignored. The setting Enforce email privacy settings is ignored.
- **FILTER**—No email is sent to recipients that have the Email Opt Out option set. Emails are sent to the other recipients. The setting Enforce email privacy settings is ignored.
- **REJECT**—If any of the recipients have the Email Opt Out option set, `sendEmail()` throws an error and no email is sent. The setting Enforce email privacy settings is respected, as are the selections in the data privacy record based on the Individual object. If any of the recipients have Don’t Market, Don’t Process, or Forget This Individual selected, `sendEmail()` throws an error and no email is sent.

**Return Value**

Type: `void`

**Example**

This example shows how to send an email with the opt-out setting enforced. Recipients are specified by their IDs. The FILTER option causes the email to be sent only to recipients that haven’t opted out from email. This example uses dot notation of the email properties, which is equivalent to using the set methods.

```java
Messaging.SingleEmailMessage message = new Messaging.SingleEmailMessage();  
// Set recipients to two contact IDs.
```
// Replace IDs with valid record IDs in your org.
message.toAddresses = new String[] { '003D000000QDexS', '003D000000QDfW5' };
message.optOutPolicy = 'FILTER';
message.subject = 'Opt Out Test Message';
message_plainTextBody = 'This is the message body.';
Messaging.SingleEmailMessage[] messages =
    new List<Messaging.SingleEmailMessage> {message};
Messaging.SendEmailResult[] results = Messaging.sendEmail(messages);
if (results[0].success) {
    System.debug('The email was sent successfully.');
} else {
    System.debug('The email failed to send: '
        + results[0].errors[0].message);
}

setPlainTextBody(plainTextBody)
Optional. The text version of the email, specified by the sender. Specify a value for setTemplateId, setHtmlBody, or setPlainTextBody. Or, you can define both setHtmlBody and setPlainTextBody.

Signature
public Void setPlainTextBody(String plainTextBody)

Parameters
plainTextBody
    Type: String

Return Value
Type: Void

setOrgWideEmailAddressId(emailAddressId)
Optional. The ID of the organization-wide email address associated with the outgoing email. If you're using Apex to send emails from the guest user, set the sender to the verified org-wide email address or the emails are blocked. The object's DisplayName field cannot be set if the setSenderDisplayName field is already set.

Signature
public Void setOrgWideEmailAddressId(ID emailAddressId)

Parameters
emailAddressId
    Type: ID
Usage
After you create an org-wide email address, you're sent a confirmation email to verify it. Copy the Id from the URL and use the `setOrgWideEmailAddressId(Id)` method on your instance of `Messaging.SingleEmailMessage`.

To avoid hard-coding an ID, after creating your org-wide email address, you can query them.

```java
OrgWideEmailAddress[] owa = [select Id from OrgWideEmailAddress where Address = 'doNotReply@somedomain.com'];
Messaging.SingleEmailMessage mail = new Messaging.SingleEmailMessage();
if ( owa.size() > 0 ) {
    mail.setOrgWideEmailAddressId(owa.get(0).Id);
}
```

Return Value
Type: Void

`setReferences(references)`
Optional. The References field of the outgoing email. Identifies an email thread. Contains the parent emails’ References and message IDs, and possibly the In-Reply-To fields.

**Signature**

```java
public Void setReferences(String references)
```

**Parameters**

`references`
Type: `String`

Return Value
Type: Void

`setSubject(subject)`
Optional. The email subject line. If you are using an email template, the subject line of the template overrides this value.

**Signature**

```java
public Void setSubject(String subject)
```

**Parameters**

`subject`
Type: `String`

Return Value
Type: Void
**setTargetObjectId(targetObjectId)**

Required if using a template, optional otherwise. The ID of the contact, lead, or user to which the email will be sent. The ID you specify sets the context and ensures that merge fields in the template contain the correct data.

**Signature**

```java
public Void setTargetObjectId(ID targetObjectId)
```

**Parameters**

- **targetObjectId**
  - Type: ID

**Return Value**

Type: Void

**Usage**

Do not specify the IDs of records that have the Email Opt Out option selected.

All emails must have a recipient value in at least one of the following fields:

- toAddresses
- ccAddresses
- bccAddresses
- targetObjectId

**setTemplateId(templateId)**

Required if using a template, optional otherwise. The ID of the template used to create the email.

**Signature**

```java
public Void setTemplateId(ID templateId)
```

**Parameters**

- **templateId**
  - Type: ID

**Return Value**

Type: Void

**setToAddresses(toAddresses)**

Optional. A list of email addresses or object IDs of the contacts, leads, and users you’re sending the email to. The maximum size for this field is 4,000 bytes. The maximum total of toAddresses, ccAddresses, and bccAddresses per email is 150. All recipients in these three fields count against the limit for email sent using Apex or the API.
**Signature**

```java
public Void setToAddresses(String[] toAddresses)
```

**Parameters**

toAddresses  
Type: `String[]`

**Return Value**

Type: Void

**Usage**

All emails must have a recipient value in at least one of the following fields:

- toAddresses
- ccAddresses
- bccAddresses
- targetObjectId

**setTreatBodiesAsTemplate(treatAsTemplate)**

Optional. If set to `true`, the subject, plain text, and HTML text bodies of the email are treated as template data. The merge fields are resolved using the `renderEmailTemplate` method. Default is `false`.

**Signature**

```java
public void setTreatBodiesAsTemplate(Boolean treatAsTemplate)
```

**Parameters**

treatAsTemplate  
Type: `Boolean`

**Return Value**

Type: void

**setTreatTargetObjectAsRecipient(treatAsRecipient)**

Optional. If set to `true`, the `targetObjectId` (a contact, lead, or user) is the recipient of the email. If set to `false`, the `targetObjectId` is supplied as the `WhoId` field for template rendering but isn't a recipient of the email. The default is `true`.

**Signature**

```java
public void setTreatTargetObjectAsRecipient(Boolean treatAsRecipient)
```
Parameters

treatAsRecipient

Type: Boolean

Return Value

Type: void

Usage

Note: You can set TO, CC, and BCC addresses using the email messaging methods regardless of whether a template is used for the email or the target object is a recipient.

setWhatId(whatId)

If you specify a contact for the targetObjectId field, you can specify an optional whatId as well. This helps to further ensure that merge fields in the template contain the correct data.

Signature

public Void setWhatId(ID whatId)

Parameters

whatId

Type: ID

Return Value

Type: Void

Usage

The value must be one of the following types:

- Account
- Asset
- Campaign
- Case
- Contract
- Opportunity
- Order
- Product
- Solution
- Custom
The `Metadata` namespace provides classes and methods for working with custom metadata in Salesforce. Salesforce uses metadata types and components to represent org configuration and customization. Metadata is used for org settings that admins control or configuration information applied by installed apps and packages. Use the classes in the `Metadata` namespace to access metadata from within Apex code.

Metadata access in Apex is available for Apex classes using API version 40.0 and later.

For more information, see `Metadata`.

The following are the classes in the `Metadata` namespace.

IN THIS SECTION:

- `AnalyticsCloudComponentLayoutItem Class`
  Represents the settings for a Wave Analytics dashboard on a standard or custom page.

- `ConsoleComponent Class`
  Represents a custom console component on a section of a page layout.

- `Container Class`
  Represents a location and style in which to display more than one custom console component in the sidebars of the console.

- `CustomConsoleComponents Class`
  Represents custom console components (Visualforce pages, lookup fields, or related lists) on a page layout.

- `CustomMetadata Class`
  Represents records of custom metadata types.

- `CustomMetadataValue Class`
  Represents custom metadata values for a custom metadata component.

- `DeployCallback Interface`
  An interface for metadata deployment callback classes.

- `DeployCallbackContext Class`
  Represents context information for a deployment job.

- `DeployContainer Class`
  Represents a container for custom metadata components to be deployed.

- `DeployDetails Class`
  Contains detailed information on deployed components.

- `DeployMessage Class`
  Represents result information for the deployment of a metadata component.

- `DeployProblemTypeEnum`
  Describes the problem type for an unsuccessful component deploy.

- `DeployResult Class`
  Represents the results of a metadata deployment.

- `DeployStatus Enum`
  The result status of a deployment.
FeedItemTypeEnum Enum
The type of feed item in a feed-based page layout.

FeedLayout Class
Represents the values that define the feed view of a feed-based page layout. Feed-based layouts are available on Account, Case, Contact, Lead, Opportunity, custom, and external objects. They include a feed view and a detail view.

FeedLayoutComponent Class
Represents a component in the feed view of a feed-based page layout.

FeedLayoutComponentTypeEnum Enum
Indicates the type of feed layout component.

FeedLayoutFilter Class
Represents a feed filter option in the feed view of a feed-based page layout. A filter can have only standardFilter or feedItemType set.

FeedLayoutFilterPosition Enum
Describes where the feed filters list is included in the layout.

FeedLayoutFilterType Enum
The type of feed layout filter.

Layout Class
Represents the metadata associated with a page layout.

LayoutColumn Class
Represents the items in a column within a layout section.

LayoutHeader Enum
Represents tagging types used for Metadata.Layout.headers

LayoutItem Class
Represents the valid values that define a layout item.

LayoutSection Class
Represents a section of a page layout, such as the Custom Links section.

LayoutSectionStyle Enum
Describes the possible styles for a layout section.

Metadata Class
An abstract base class that represents a custom metadata component.

MetadataTypeEnum Enum
Represents the custom metadata components available in Apex.

MetadataValue Class
An abstract base class that represents a custom metadata component field.

MiniLayout Class
Represents a mini view of a record in the Console tab, hover details, and event overlays.

Operations Class
Represents a class to execute metadata operations, such as retrieving or deploying custom metadata.

PlatformActionList Class
Represents the list of actions, and their order, that display in the Salesforce mobile action bar for the layout.
PlatformActionListContextEnum Enum
Describes the different contexts of action lists.

PlatformActionListItem Class
Represents an action in the platform action list for a layout.

PlatformActionTypeEnum Enum
The type of action for a PlatformActionListItem.

PrimaryTabComponents Class
Represents custom console components on primary tabs in the Salesforce console.

QuickActionList Class
Represents the list of actions associated with the page layout.

QuickActionListItem Class
Represents an action in the QuickActionList.

RelatedContent Class
Represents the Mobile Cards section of the page layout.

RelatedContentItem Class
Represents an individual item in the RelatedContent list.

RelatedList Class
Represents related list custom components on the sidebars of the Salesforce console.

RelatedListItem Class
Represents an item in the related list in a page layout.

ReportChartComponentLayoutItem Class
Represents the settings for a report chart on a standard or custom page.

ReportChartComponentSize Enum
Describes the size of the displayed report chart component.

SidebarComponent Class
Represents a specific custom console component to display in a container that hosts multiple components in one of the sidebars of the Salesforce console.

SortOrder Enum
Describes the sort order of a related list.

StatusCode Enum
Describes the status code for an unsuccessful component deploy.

SubtabComponents Class
Represents custom console components on subtabs in the Salesforce console.

SummaryLayoutStyleEnum Enum
Describes the highlights panel style for a SummaryLayout.

SummaryLayout Class
Controls the appearance of the highlights panel, which summarizes key fields in a grid at the top of a page layout, when Case Feed is enabled.

SummaryLayoutItem Class
Controls the appearance of an individual field and its column and row position within the highlights panel grid, when Case Feed is enabled. You can have two fields per each grid in a highlights panel.
UiBehavior Enum
Describes the behavior for a layout item on a layout page.

AnalyticsCloudComponentLayoutItem Class
Represents the settings for a Wave Analytics dashboard on a standard or custom page.

Namespace
Metadata

Usage
Use this class when accessing Metadata.Layout metadata components. For more information, see “AnalyticsCloudComponentLayoutItem” in the Metadata API Developer Guide.

IN THIS SECTION:
AnalyticsCloudComponentLayoutItem Properties
AnalyticsCloudComponentLayoutItem Methods

AnalyticsCloudComponentLayoutItem Properties
The following are properties for AnalyticsCloudComponentLayoutItem.

IN THIS SECTION:
assetType
Specifies the type of Wave Analytics asset.
devName
Unique development name of the dashboard to add.
error
An error string that is populated only when an error occurred in the underlying dashboard.
filter
Dashboard filters for mapping data fields in the dashboard to the object’s fields.
height
Specifies the height of the dashboard, in pixels.
hideOnError
Controls whether users see a dashboard that has an error.
showHeader
If true, includes the dashboard’s header bar. If false, the dashboard appears without a header bar.
showSharing
If set to true, and the dashboard is shareable the dashboard shows the Share icon. If set to false, the dashboard doesn’t show the Share icon.
showTitle
If true, includes the dashboard’s title above the dashboard. If false, the dashboard appears without a title.

width
Specifies the width of the dashboard, in pixels or percentage.

**assetType**
Specifies the type of Wave Analytics asset.

**Signature**
```
public String assetType {get; set;}
```

**Property Value**
Type: `String`

**devName**
Unique development name of the dashboard to add.

**Signature**
```
public String devName {get; set;}
```

**Property Value**
Type: `String`

**error**
An error string that is populated only when an error occurred in the underlying dashboard.

**Signature**
```
public String error {get; set;}
```

**Property Value**
Type: `String`

**filter**
Dashboard filters for mapping data fields in the dashboard to the object’s fields.

**Signature**
```
public String filter {get; set;}
```
Property Value
Type: String

**height**
Specifies the height of the dashboard, in pixels.

Signature
```java
public Integer height {get; set;}
```

Property Value
Type: Integer

**hideOnError**
Controls whether users see a dashboard that has an error.

Signature
```java
public Boolean hideOnError {get; set;}
```

Property Value
Type: Boolean

**showHeader**
If true, includes the dashboard's header bar. If false, the dashboard appears without a header bar.

Signature
```java
public Boolean showHeader {get; set;}
```

Property Value
Type: Boolean

**showSharing**
If set to true, and the dashboard is shareable the dashboard shows the Share icon. If set to false, the dashboard doesn’t show the Share icon.

Signature
```java
public Boolean showSharing {get; set;}
```
Property Value
Type: Boolean

**showTitle**
If true, includes the dashboard’s title above the dashboard. If false, the dashboard appears without a title.

**Signature**
```java
public Boolean showTitle {get; set;}
```

Property Value
Type: Boolean

**width**
Specifies the width of the dashboard, in pixels or percentage.

**Signature**
```java
public String width {get; set;}
```

Property Value
Type: String

**AnalyticsCloudComponentLayoutItem Methods**
The following are methods for AnalyticsCloudComponentLayoutItem.

**IN THIS SECTION:**
- **clone()**
  Makes a duplicate copy of the Metadata.AnalyticsCloudComponentLayoutItem.

**clone()**
Makes a duplicate copy of the Metadata.AnalyticsCloudComponentLayoutItem.

**Signature**
```java
public Object clone()
```

**Return Value**
Type: Object
ConsoleComponent Class

Represents a custom console component on a section of a page layout.

Namespace

Metadata

Usage

Use this class when accessing Metadata.Layout metadata components. For more information, see “ConsoleComponent” in the Metadata API Developer Guide.

IN THIS SECTION:

ConsoleComponent Properties

ConsoleComponent Methods

ConsoleComponent Properties

The following are properties for ConsoleComponent.

IN THIS SECTION:

height
The height of the custom console component in pixels.

location
The location of the custom console component on the page layout. Valid values are right, left, top, and bottom.

visualforcePage
The unique name of the custom console component.

width
The width of the custom console component in pixels.

height

The height of the custom console component in pixels.

Signature

public Integer height {get; set;}

Property Value

Type: Integer

location

The location of the custom console component on the page layout. Valid values are right, left, top, and bottom.
Signature
public String location {get; set;}

Property Value
Type: String

visualforcePage
The unique name of the custom console component.

Signature
public String visualforcePage {get; set;}

Property Value
Type: String

width
The width of the custom console component in pixels.

Signature
public Integer width {get; set;}

Property Value
Type: Integer

ConsoleComponent Methods
The following are methods for ConsoleComponent.

IN THIS SECTION:
  clone()
    Makes a duplicate copy of the Metadata.ConsoleComponent.

clone()
Makes a duplicate copy of the Metadata.ConsoleComponent.

Signature
public Object clone()
Return Value
Type: Object

Container Class
Represents a location and style in which to display more than one custom console component in the sidebars of the console.

Namespace
Metadata

Usage
Use this class when accessing `Metadata.Layout` metadata components. For more information, see “Container” in the `Metadata API Developer Guide`.

IN THIS SECTION:
- Container Properties
- Container Methods

Container Properties
The following are properties for `Container`.

IN THIS SECTION:
- `height`
- `isContainerAutoSizeEnabled`
- `region`
- `sidebarComponents`
- `style`
- `unit`
- `width`

**height**
The height of the component’s container. The `unit` property determines the unit of measurement, in pixels or percent.

**isContainerAutoSizeEnabled**
If set to true, stacked console components in the sidebars autosize vertically.

**region**
The location of the component’s container (right, left, bottom, top).

**sidebarComponents**
Represents a specific custom console component to display in the components’ container.

**style**
The style of the container in which to display multiple components (stack, tab, accordion).

**unit**
The unit of measurement, in pixels or percent, for the height or width of the components’ container.

**width**
The width of the component’s container. The `unit` property determines the unit of measurement, in pixels or percent.
**Signature**

```java
public Integer height {get; set;}
```

**Property Value**

Type: `Integer`

---

**isContainerAutoSizeEnabled**

If set to true, stacked console components in the sidebars autosize vertically.

**Signature**

```java
public Boolean isContainerAutoSizeEnabled {get; set;}
```

**Property Value**

Type: `Boolean`

---

**region**

The location of the component's container (right, left, bottom, top).

**Signature**

```java
public String region {get; set;}
```

**Property Value**

Type: `String`

---

**sidebarComponents**

Represents a specific custom console component to display in the components' container.

**Signature**

```java
public List<Metadata.SidebarComponent> sidebarComponents {get; set;}
```

**Property Value**

Type: `List<Metadata.SidebarComponent>`

---

**style**

The style of the container in which to display multiple components (stack, tab, accordion).

**Signature**

```java
public String style {get; set;}
```
Property Value
Type: `String`

**unit**
The unit of measurement, in pixels or percent, for the height or width of the components’ container.

**Signature**
`public String unit {get; set;}`

Property Value
Type: `String`

**width**
The width of the component’s container. The `unit` property determines the unit of measurement, in pixels or percent.

**Signature**
`public Integer width {get; set;}`

Property Value
Type: `Integer`

**Container Methods**
The following are methods for `Container`.

**IN THIS SECTION:**
`clone()`  
Makes a duplicate copy of the `Metadata.Container`.

`clone()`  
Makes a duplicate copy of the `Metadata.Container`.

**Signature**
`public Object clone()`  

**Return Value**
Type: `Object`
CustomConsoleComponents Class

Represents custom console components (Visualforce pages, lookup fields, or related lists) on a page layout.

Namespace

Metadata

Usage

Use this class when accessing Metadata.Layout metadata components. For more information, see “CustomConsoleComponents” in the Metadata API Developer Guide.

In this section:

- CustomConsoleComponents Properties
- CustomConsoleComponents Methods

CustomConsoleComponents Properties

The following are properties for CustomConsoleComponents.

In this section:

- primaryTabComponents
- subtabComponents

primaryTabComponents

Represents custom console components on primary tabs in the Salesforce console.

Signature

```java
public Metadata.PrimaryTabComponents primaryTabComponents {get; set;}
```

Property Value

Type: Metadata.PrimaryTabComponents

subtabComponents

Represents custom console components on subtabs in the Salesforce console.

Signature

```java
public Metadata.SubtabComponents subtabComponents {get; set;}
```
CustomConsoleComponents Methods

The following are methods for CustomConsoleComponents.

**IN THIS SECTION:**

**clone()**

Makes a duplicate copy of the Metadata.CustomConsoleComponents.

**Signature**

```java
public Object clone()
```

**Return Value**

Type: Object

CustomMetadata Class

Represents records of custom metadata types.

**Warning:** Protected custom metadata types behave like public custom metadata types when they are outside of a managed package. Public custom metadata types are readable for all profiles, including the guest user. Do not store secrets, personally identifying information, or any private data in these records. Use protected custom metadata types only in managed packages. Outside of a managed package, use named credentials or encrypted custom fields to store secrets like OAuth tokens, passwords, and other confidential material.

**Namespace**

Metadata

**Usage**

Use Metadata.CustomMetadata to represent records of custom metadata types in Apex. For more information, see Custom Metadata Types in the Metadata API Developer Guide.

**Example**

```java
// Set up custom metadata to be created in the subscriber org.
Metadata.CustomMetadata customMetadata = new Metadata.CustomMetadata();
customMetadata.fullName = 'ISVNamespace__MetadataTypeName.MetadataRecordName';
```
customField = new Metadata.CustomMetadataValue();
customField.field = 'customField__c';
customField.value = 'New value';
customMetadata.values.add(customField);

Note: When you assign namespaces to records, provide full, qualified record names to the app. If both the type and the record are in Namespace, use something like: customMetadata.fullName = 'Namespace__MetadataTypeName.Namespace__MetadataRecordName'

IN THIS SECTION:

CustomMetadata Properties

CustomMetadata Methods

CustomMetadata Properties

The following are properties for CustomMetadata.

IN THIS SECTION:

description
The description of the custom metadata.

label
The label of the custom metadata record.

protected_x
Property that describes whether the custom metadata record is a protected component.

values
A list of custom metadata values, such as custom fields, for the custom metadata record.

description

The description of the custom metadata.

Signature

public String description {get; set;}

Property Value

Type: String

label

The label of the custom metadata record.

Signature

public String label {get; set;}
Property Value
Type: String

**protected_x**
Property that describes whether the custom metadata record is a protected component.

Signature
```csharp
public Boolean protected_x {get; set;}
```

Property Value
Type: Boolean

**values**
A list of custom metadata values, such as custom fields, for the custom metadata record.

Signature
```csharp
public List<Metadata.CustomMetadataValue> values {get; set;}
```

Property Value
Type: List<Metadata.CustomMetadataValue>

**CustomMetadata Methods**
The following are methods for CustomMetadata.

IN THIS SECTION:

**clone()**
Makes a duplicate copy of the Metadata.CustomMetadata.

**clone()**
Makes a duplicate copy of the Metadata.CustomMetadata.

Signature
```csharp
public Object clone()
```

Return Value
Type: Object
CustomMetadataValue Class

Represents custom metadata values for a custom metadata component.

Namespace

Metadata

Usage

Use Metadata CustomMetadataValue to access values for custom fields of custom metadata records.

Supported Apex primitive types are:

- Boolean
- Date
- DateTime
- Decimal
- Double
- Integer
- Long
- String

Example

```java
// Set a custom field value for a custom metadata record
Metadata.CustomMetadataValue customField = new Metadata.CustomMetadataValue();
customField.field = 'CustomField1__c';
customField.value = 'New Value';
customMetadata.values.add(customField);
```

IN THIS SECTION:

- CustomMetadataValue Properties
- CustomMetadataValue Methods

CustomMetadataValue Properties

The following are properties for CustomMetadataValue.

IN THIS SECTION:

- field
  - The field name for the custom metadata value.
- value
  - The field value for the custom metadata value.
**field**
The field name for the custom metadata value.

**Signature**
`public String field {get; set;}

**Property Value**
Type: String

**value**
The field value for the custom metadata value.

**Signature**
`public Object value {get; set;}

**Property Value**
Type: Object
Supported Apex primitive types are:
- Boolean
- Date
- DateTime
- Decimal
- Double
- Integer
- Long
- String

When setting the value for relationship fields, use the qualified API name of the related metadata, not the ID. For more information, see Primitive Data Types.

**CustomMetadataValue Methods**
The following are methods for CustomMetadataValue.

IN THIS SECTION:
- `clone()`
  - Makes a duplicate copy of the Metadata.CustomMetadataValue.

- `clone()`
  - Makes a duplicate copy of the Metadata.CustomMetadataValue.
DeployCallback Interface

An interface for metadata deployment callback classes.

Namespace

Metadata

Usage

You must provide a callback class for the asynchronous deployment of custom metadata through Apex. This class must implement the `Metadata.DeployCallback` interface.

Salesforce calls your `DeployCallback.handleResult(var1, var2)` method asynchronously once the queued deployment completes. Because the callback is called as asynchronous Apex after deployment, there may be a brief period where the deploy has completed, but your callback has not been called yet.

IN THIS SECTION:

- DeployCallback Methods
- DeployCallback Example Implementation

DeployCallback Methods

The following are methods for `DeployCallback`.

IN THIS SECTION:

- `handleResult(var1, var2)`
  
  Method that is called when the asynchronous deployment of custom metadata completes.

`handleResult(var1, var2)`

Method that is called when the asynchronous deployment of custom metadata completes.

Signature

```
public void handleResult(Metadata.DeployResult var1, Metadata.DeployCallbackContext var2)
```
Parameters

var1
Type: Metadata.DeployResult
The results of the asynchronous deployment.

var2
Type: Metadata.DeployCallbackContext
The context for the queued asynchronous deployment job.

Return Value
Type: void

DeployCallback Example Implementation

This is an example implementation of the Metadata.DeployCallback interface.

```java
public class MyCallback implements Metadata.DeployCallback {
    public void handleResult(Metadata.DeployResult result,
                              Metadata.DeployCallbackContext context) {
        if (result.status == Metadata.DeployStatus.Succeeded) {
            // Deployment was successful
        } else {
            // Deployment was not successful
        }
    }
}
```

The following example uses this implementation for a deployment.

```java
// Setup callback and deploy
MyCallback callback = new MyCallback();
Metadata.Operations.enqueueDeployment(mdContainer, callback);
```

DeployCallbackContext Class

Represents context information for a deployment job.

Namespace

Metadata

Usage

After an asynchronous metadata deployment finishes, Salesforce provides an instance of Metadata.DeployCallbackContext in an asynchronous call to your implementation of handleResult() in your Metadata.DeployCallback class.
Example

```java
public void handleResult(Metadata.DeployResult result,
    Metadata.DeployCallbackContext context) {
    // Check the callback job ID for the deployment
    Id jobId = context.getCallbackJobId();
    // ...process the results...
}
```

IN THIS SECTION:
DeployCallbackContext Methods

DeployCallbackContext Methods
The following are methods for DeployCallbackContext.

IN THIS SECTION:
clone()  
Makes a duplicate copy of the Metadata.DeployCallbackContext.

callbackJobId()  
Gets the asynchronous Apex job ID for the callback job.

clone()

Makes a duplicate copy of the Metadata.DeployCallbackContext.

**Signature**

```java
public Object clone()
```

**Return Value**

Type: Object

getCallbackJobId()

Gets the asynchronous Apex job ID for the callback job.

**Signature**

```java
public Id getCallbackJobId()
```

**Return Value**

Type: Id
DeployContainer Class

Represents a container for custom metadata components to be deployed.

Namespace

Metadata

Usage

Use Metadata.DeployContainer to manage custom metadata components for deployment. A container must have one or more components before being deployed.

Example

```java
// Use DeployContainer for deployment
Metadata.DeployContainer mdContainer = new Metadata.DeployContainer();
mdContainer.addMetadata(customMetadata);
...

// Enqueue deploy
Metadata.Operations.enqueueDeployment(mdContainer, callback);
```

IN THIS SECTION:

DeployContainer Methods

DeployContainer Methods

The following are methods for DeployContainer.

IN THIS SECTION:

addMetadata(md)

Add a custom metadata component to the container.

clonel)

Makes a duplicate copy of the Metadata.DeployContainer.

getMetadata()

Retrieves a list of custom metadata components from the container.

removeMetadata(md)

Removes a metadata component from the container.

removeMetadataByFullName(fullName)

Removes a metadata component from the container using the component’s full name.

addMetadata (md)

Add a custom metadata component to the container.
Signature
public void addMetadata(Metadata.Metadata md)

Parameters
md
Type: Metadata.Metadata
A custom metadata component class that derives from Metadata.Metadata. Avoid adding components to a Metadata.DeployContainer that have the same Metadata.Metadata.fullName because it causes deployment errors.

Return Value
Type: void

clone()
Makes a duplicate copy of the Metadata.DeployContainer.

Signature
public Object clone()

Return Value
Type: Object

getMetadata()
Retrieves a list of custom metadata components from the container.

Signature
public List<Metadata.Metadata> getMetadata()

Return Value
Type: List<Metadata.Metadata>

removeMetadata(md)
Removes a metadata component from the container.

Signature
public Boolean removeMetadata(Metadata.Metadata md)
DeployDetails Class

Contains detailed information on deployed components.

Namespace

Metadata

Usage

Use this class to obtain a list of the successfully and unsuccessfully deployed components after a completed deployment by Salesforce in your Metadata.DeployCallback results.

IN THIS SECTION:

DeployDetails Properties
DeployDetails Methods
DeployDetails Properties

The following are properties for DeployDetails.

IN THIS SECTION:

- **componentFailures**
  Contains a list of information about components that failed to deploy.

- **componentSuccesses**
  Contains a list of information about components that deployed successfully.

**componentFailures**

Contains a list of information about components that failed to deploy.

**Signature**

```csharp
public List<Metadata.DeployMessage> componentFailures {get; set;}
```

**Property Value**

Type: `List<Metadata.DeployMessage>`

**componentSuccesses**

Contains a list of information about components that deployed successfully.

**Signature**

```csharp
public List<Metadata.DeployMessage> componentSuccesses {get; set;}
```

**Property Value**

Type: `List<Metadata.DeployMessage>`

DeployDetails Methods

The following are methods for DeployDetails.

IN THIS SECTION:

- **clone()**
  Makes a duplicate copy of the `Metadata.DeployDetails`.

**clone()**

Makes a duplicate copy of the `Metadata.DeployDetails`.
Signature

```java
public Object clone()
```

Return Value

Type: Object

DeployMessage Class

Represents result information for the deployment of a metadata component.

Namespace

`Metadata`

Usage

Use `DeployMessage` to access detailed information about component deployments. Salesforce provides a list of `DeployMessages` for a completed deployment via the `DeployDetails` and `DeployResults` instances sent in the `DeployCallback.handleResult()` callback.

IN THIS SECTION:

- DeployMessage Properties
- DeployMessage Methods

DeployMessage Properties

The following are properties for `DeployMessage`.

IN THIS SECTION:

- `changed`
  Determines whether the component was changed after deployment. If true, the component was changed as a result of the deployment. If false, the deployed component was the same as the corresponding component already in the org.
- `columnNameNumber`
  Each component is represented by a text file. If an error occurs during deployment, this property represents the column of the text file where the error occurred.
- `componentType`
  The metadata type of the component in the deployment.
- `created`
  If true, the component was created as a result of the deployment. If false, the component was modified as a result of the deployment.
- `createdDate`
  The date and time when the component was created as a result of the deployment.
deleted
If true, the component was deleted as a result of the deployment. If false, the component was either new or modified as result of the deployment.

fileName
The name of the file in the metadata archive used to deploy the component.

fullName
Full name for the custom metadata component.

id
ID of the component that was deployed.

lineNumber
Each component is represented by a text file. If an error occurs during deployment, this field represents the line number of the text file where the error occurred.

problem
If an error or warning occurred, this field contains a description of the problem that caused the deployment to fail.

problemType
Indicates the problem type, for example, an error or warning.

success
Indicates whether the component was successfully deployed (true) or not (false).

changed
Determines whether the component was changed after deployment. If true, the component was changed as a result of the deployment. If false, the deployed component was the same as the corresponding component already in the org.

Signature

```java
public Boolean changed {get; set;}
```

Property Value
Type: Boolean

columnNumber
Each component is represented by a text file. If an error occurs during deployment, this property represents the column of the text file where the error occurred.

Signature

```java
public Integer columnNumber {get; set;}
```

Property Value
Type: Integer
componentType
The metadata type of the component in the deployment.

Signature
public String componentType {get; set;}

Property Value
Type: String

created
If true, the component was created as a result of the deployment. If false, the component was modified as a result of the deployment.

Signature
public Boolean created {get; set;}

Property Value
Type: Boolean

createdDate
The date and time when the component was created as a result of the deployment.

Signature
public Datetime createdDate {get; set;}

Property Value
Type: Datetime

deleted
If true, the component was deleted as a result of the deployment. If false, the component was either new or modified as result of the deployment.

Signature
public Boolean deleted {get; set;}

Property Value
Type: Boolean
**fileName**
The name of the file in the metadata archive used to deploy the component.

**Signature**
```java
public String fileName {get; set;}
```

**Property Value**
Type: String

**fullName**
Full name for the custom metadata component.

**Signature**
```java
public String fullName {get; set;}
```

**Property Value**
Type: String

**id**
ID of the component that was deployed.

**Signature**
```java
public Id id {get; set;}
```

**Property Value**
Type: Id

**lineNumber**
Each component is represented by a text file. If an error occurs during deployment, this field represents the line number of the text file where the error occurred.

**Signature**
```java
public Integer lineNumber {get; set;}
```

**Property Value**
Type: Integer
**problem**
If an error or warning occurred, this field contains a description of the problem that caused the deployment to fail.

**Signature**
```csharp
public String problem {get; set;}
```

**Property Value**
Type: `String`

**problemType**
Indicates the problem type, for example, an error or warning.

**Signature**
```csharp
public Metadata.DeployProblemType problemType {get; set;}
```

**Property Value**
Type: `Metadata.DeployProblemType`

**success**
Indicates whether the component was successfully deployed (true) or not (false).

**Signature**
```csharp
public Boolean success {get; set;}
```

**Property Value**
Type: `Boolean`

**DeployMessage Methods**
The following are methods for `DeployMessage`.

**IN THIS SECTION:**

2383
DeployProblemType Enum

Describes the problem type for an unsuccessful component deploy.

Enum Values

The following are the values of the Metadata_DeployProblemType enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error</td>
<td>The deploy problem is an error.</td>
</tr>
<tr>
<td>Info</td>
<td>The deploy problem is of type &quot;Info&quot;.</td>
</tr>
<tr>
<td>Warning</td>
<td>The deploy problem is a warning.</td>
</tr>
</tbody>
</table>

SEE ALSO:

StatusCode Enum

DeployResult Class

Represents the results of a metadata deployment.

Namespace

Metadata

Usage

After an asynchronous metadata deployment finishes, Salesforce provides an instance of Metadata_DeployResult in a call to your implementation of handleResult() in your Metadata_DeployCallback class.

Example

```plaintext
public void handleResult(Metadata.DeployResult result,
            Metadata.DeployCallbackContext context) {
    if (result.status == Metadata.DeployStatus.Succeeded) {
        // Deployment was successful
    } else {
        // Deployment was not successful
    }
```
The following are properties for `DeployResult`.

- **canceledBy**: ID of the user who canceled the queued deployment.
- **canceledByName**: Full name of the user who canceled the queued deployment.
- **checkOnly**: Indicates whether the deployment checked only the validity of the deployed files without making changes in the org. A check-only deployment does not deploy components or change the org in any way.
- **completedDate**: Date and time for when the deployment process ended.
- **createdBy**: ID of the user who created the deployment job.
- **createdByName**: Full name of the user who created the deployment job.
- **createdDate**: Date and time the deployment job was first queued.
- **details**: Provides the details for components in a completed deployment.
- **done**: Indicates whether Salesforce finished processing the deployment.
- **errorMessage**: Message corresponding to the values in the `errorStatusCode` property, if any.
- **errorStatusCode**: If an error occurs during deployment, a status code is returned. The message corresponding to the status code is returned in the `errorMessage` property.
- **id**: ID of the deployment job.
- **ignoreWarnings**: Specifies whether a deployment continues, even if the deployment generates warnings.
lastModifiedDate
Date and time of the last update for the deployment process.
messages
A list of all the detail messages for a deployment.
numberComponentErrors
The number of components that generated errors during the deployment.
numberComponentsDeployed
The number of components deployed in the deployment process. Use this value with the numberComponentsTotal property to get an estimate of the deployment’s progress.
numberComponentsTotal
The total number of components in the deployment. Use this value with the numberComponentsDeployed property to get an estimate of the deployment’s progress.
rollbackOnError
Indicates whether any failure causes a complete rollback (true) or not (false) of the deployment.
startDate
Date and time the deployment process began.
stateDetail
Indicates which component is being deployed.
status
Indicates the current state of the deployment.
success
Indicates whether the deployment was successful (true) or not (false).

canceledBy
ID of the user who canceled the queued deployment.

Signature
public String canceledBy {get; set;}

Property Value
Type: String

canceledByName
Full name of the user who canceled the queued deployment.

Signature
public String canceledByName {get; set;}

Property Value
Type: String
checkOnly
Indicates whether the deployment checked only the validity of the deployed files without making changes in the org. A check-only deployment does not deploy components or change the org in any way.

Signature
public Boolean checkOnly {get; set;}

Property Value
Type: Boolean

completedDate
Date and time for when the deployment process ended.

Signature
public Datetime completedDate {get; set;}

Property Value
Type: Datetime

createdBy
ID of the user who created the deployment job.

Signature
public String createdBy {get; set;}

Property Value
Type: String

createdByName
Full name of the user who created the deployment job.

Signature
public String createdByName {get; set;}

Property Value
Type: String
**createdDate**
Date and time the deployment job was first queued.

**Signature**
```
public Datetime createdDate {get; set;}
```

**Property Value**
Type: Datetime

**details**
Provides the details for components in a completed deployment.

**Signature**
```
public Metadata.DeployDetails details {get; set;}
```

**Property Value**
Type: Metadata.DeployDetails

**done**
Indicates whether Salesforce finished processing the deployment.

**Signature**
```
public Boolean done {get; set;}
```

**Property Value**
Type: Boolean

**errorMessage**
Message corresponding to the values in the errorStatusCode property, if any.

**Signature**
```
public String errorMessage {get; set;}
```

**Property Value**
Type: String
**errorStatusCode**

If an error occurs during deployment, a status code is returned. The message corresponding to the status code is returned in the `errorMessage` property.

**Signature**

```java
public String errorStatusCode {get; set;}
```

**Property Value**

Type: `String`

For a description of each status code value, see Core Data Types Used in API Calls in the SOAP API Developer Guide.

**id**

ID of the deployment job.

**Signature**

```java
public Id id {get; set;}
```

**Property Value**

Type: `Id`

**ignoreWarnings**

Specifies whether a deployment continues, even if the deployment generates warnings.

**Signature**

```java
public Boolean ignoreWarnings {get; set;}
```

**Property Value**

Type: `Boolean`

**lastModifiedDate**

Date and time of the last update for the deployment process.

**Signature**

```java
public Datetime lastModifiedDate {get; set;}
```

**Property Value**

Type: `Datetime`
**messages**
A list of all the detail messages for a deployment.

**Signature**
```csharp
public List<Metadata.DeployMessage> messages {get; set;}
```

**Property Value**
Type: `List<Metadata.DeployMessage>`

---

**numberComponentErrors**
The number of components that generated errors during the deployment.

**Signature**
```csharp
public Integer numberComponentErrors {get; set;}
```

**Property Value**
Type: `Integer`

---

**numberComponentsDeployed**
The number of components deployed in the deployment process. Use this value with the `numberComponentsTotal` property to get an estimate of the deployment’s progress.

**Signature**
```csharp
public Integer numberComponentsDeployed {get; set;}
```

**Property Value**
Type: `Integer`

---

**numberComponentsTotal**
The total number of components in the deployment. Use this value with the `numberComponentsDeployed` property to get an estimate of the deployment’s progress.

**Signature**
```csharp
public Integer numberComponentsTotal {get; set;}
```

**Property Value**
Type: `Integer`
**rollbackOnError**
Indicates whether any failure causes a complete rollback (true) or not (false) of the deployment.

**Signature**
```
public Boolean rollbackOnError {get; set;}
```

**Property Value**
Type: Boolean

---

**startDate**
Date and time the deployment process began.

**Signature**
```
public Datetime startDate {get; set;}
```

**Property Value**
Type: Datetime

---

**stateDetail**
Indicates which component is being deployed.

**Signature**
```
public String stateDetail {get; set;}
```

**Property Value**
Type: String

---

**status**
Indicates the current state of the deployment.

**Signature**
```
public Metadata.DeployStatus status {get; set;}
```

**Property Value**
Type: Metadata.DeployStatus

---

**success**
Indicates whether the deployment was successful (true) or not (false).
Signature

```java
public Boolean success {get; set;}
```

Property Value

Type: Boolean

DeployResult Methods

The following are methods for DeployResult.

IN THIS SECTION:

- **clone()**

  Makes a duplicate copy of the Metadata.DeployResult.

**clone()**

Makes a duplicate copy of the Metadata.DeployResult.

Signature

```java
public Object clone()
```

Return Value

Type: Object

DeployStatus Enum

The result status of a deployment.

Usage

`Metadata.DeployResult.status` uses this enum to describe the results of the deployment.

Enum Values

The following are the values of the `Metadata.DeployStatus` enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Canceled</td>
<td>The queued deployment was canceled.</td>
</tr>
<tr>
<td>Canceling</td>
<td>The queued deployment is being canceled.</td>
</tr>
<tr>
<td>Failed</td>
<td>The deployment failed.</td>
</tr>
<tr>
<td>InProgress</td>
<td>The deployment has been started and is in progress.</td>
</tr>
<tr>
<td>Pending</td>
<td>The deployment has been queued but not started.</td>
</tr>
</tbody>
</table>
The deployment succeeded.

The deployment succeeded, but some components might not have been successfully deployed. Check Metadata.DeployResult for more details.

---

**FeedItemTypeEnum Enum**

The type of feed item in a feed-based page layout.

**Enum Values**

The following are the values of the Metadata.FeedItemTypeEnum enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ActivityEvent</td>
<td>Activity on tasks and events associated with a case. Available only on Case layouts.</td>
</tr>
<tr>
<td>AdvancedTextPost</td>
<td>Group announcements posted on a feed.</td>
</tr>
<tr>
<td>AnnouncementPost</td>
<td>Not used.</td>
</tr>
<tr>
<td>ApprovalPost</td>
<td>Approvals submitted on a feed.</td>
</tr>
<tr>
<td>AttachArticleEvent</td>
<td>Activity related to attaching articles to cases.</td>
</tr>
<tr>
<td>BasicTemplateFeedItem</td>
<td>Activity from the Log a Call action. Available only on layouts for objects that support Activities (tasks and events).</td>
</tr>
<tr>
<td>CallLogPost</td>
<td>Activity from the Log a Call action. Available only on layouts for objects that support Activities (tasks and events).</td>
</tr>
<tr>
<td>CanvasPost</td>
<td>Posts a canvas app makes on a feed.</td>
</tr>
<tr>
<td>CaseCommentPost</td>
<td>Activity from the Case Note action. Available only on Case layouts.</td>
</tr>
<tr>
<td>ChangeStatusPost</td>
<td>Activity from the Change Status action. Available only on Case layouts.</td>
</tr>
<tr>
<td>ChatTranscriptPost</td>
<td>Activity related to attaching Chat transcripts to cases. Available only on Case layouts.</td>
</tr>
<tr>
<td>CollaborationGroupCreated</td>
<td>Creating a public group.</td>
</tr>
<tr>
<td>CollaborationGroupUnarchived</td>
<td>Not used.</td>
</tr>
<tr>
<td>ContentPost</td>
<td>Attaching a file to a post.</td>
</tr>
<tr>
<td>CreateRecordEvent</td>
<td>Creating a record from the publisher.</td>
</tr>
<tr>
<td>DashboardComponentAlert</td>
<td>Not used.</td>
</tr>
<tr>
<td>DashboardComponentSnapshot</td>
<td>Posting a dashboard snapshot on a feed.</td>
</tr>
<tr>
<td>EmailMessageEvent</td>
<td>Activity from the Email action. Available only on Case layouts.</td>
</tr>
<tr>
<td>FacebookPost</td>
<td>Not used.</td>
</tr>
</tbody>
</table>
### FeedLayout Class

Represents the values that define the feed view of a feed-based page layout. Feed-based layouts are available on Account, Case, Contact, Lead, Opportunity, custom, and external objects. They include a feed view and a detail view.

### Namespace

*Metadata*

### Usage

Use this class when accessing `Metadata.Layout` metadata components. For more information, see “FeedLayout” in the Metadata API Developer Guide.

IN THIS SECTION:

- FeedLayout Properties
- FeedLayout Methods

### FeedLayout Properties

The following are properties for `FeedLayout`.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LinkPost</td>
<td>Attaching a URL to a post.</td>
</tr>
<tr>
<td>MilestoneEvent</td>
<td>Changing the milestone status on a case. Available only on Case layouts.</td>
</tr>
<tr>
<td>PollPost</td>
<td>Posting a poll on a feed.</td>
</tr>
<tr>
<td>ProfileSkillPost</td>
<td>Adding skills to a user's Chatter profile.</td>
</tr>
<tr>
<td>QuestionPost</td>
<td>Posting a question on a feed.</td>
</tr>
<tr>
<td>ReplyPost</td>
<td>Activity from the Portal action. Available only on Case layouts.</td>
</tr>
<tr>
<td>RypplePost</td>
<td>Creating a Thanks badge in WDC.</td>
</tr>
<tr>
<td>SocialPost</td>
<td>Activity on Twitter from the Social Post action.</td>
</tr>
<tr>
<td>TestItem</td>
<td>Creating a text post from the publisher.</td>
</tr>
<tr>
<td>TextPost</td>
<td>Making a change or group of changes to a tracked field.</td>
</tr>
<tr>
<td>TrackedChange</td>
<td>Not used.</td>
</tr>
<tr>
<td>Undefined</td>
<td>Undefined feed item.</td>
</tr>
<tr>
<td>UserStatus</td>
<td>Not used.</td>
</tr>
</tbody>
</table>
IN THIS SECTION:

- **autocollapsePublisher**
  Specifies whether the publisher is collapsed when the page loads (true) or not (false).

- **compactFeed**
  Specifies whether the feed-based page layout uses a compact feed (true) or not (false). If set to true, feed items on the page are collapsed by default, and the feed view has an updated design.

- **feedFilterPosition**
  Indicates where the feed filters list is included in the layout.

- **feedFilters**
  The individual filters displayed in the feed filters list.

- **fullWidthFeed**
  Specifies whether the feed expands horizontally to take up all available space on the page (true) or not (false).

- **hideSidebar**
  Specifies whether the sidebar is hidden (true) or not (false).

- **highlightExternalFeedItems**
  Controls whether to highlight external feed items (true) or not (false).

- **leftComponents**
  The individual components displayed in the left column of the feed view.

- **rightComponents**
  Lists the individual components displayed in the right column of the feed view.

- **useInlineFiltersInConsole**
  Indicates whether to use inline filters in the Salesforce console.

---

**autocollapsePublisher**

Specifies whether the publisher is collapsed when the page loads (true) or not (false).

**Signature**

```java
public Boolean autocollapsePublisher {get; set;}
```

**Property Value**

Type: Boolean

**compactFeed**

Specifies whether the feed-based page layout uses a compact feed (true) or not (false). If set to true, feed items on the page are collapsed by default, and the feed view has an updated design.

**Signature**

```java
public Boolean compactFeed {get; set;}
```
**Property Value**

*Type: Boolean*

**feedFilterPosition**

Indicates where the feed filters list is included in the layout.

**Signature**

```csharp
public Metadata.FeedLayoutFilterPosition feedFilterPosition {get; set;}
```

**Property Value**

*Type: FeedLayoutFilterPosition Enum*

**feedFilters**

The individual filters displayed in the feed filters list.

**Signature**

```csharp
public List<Metadata.FeedLayoutFilter> feedFilters {get; set;}
```

**Property Value**

*Type: List<FeedLayoutFilter Class>.*

**fullWidthFeed**

Specifies whether the feed expands horizontally to take up all available space on the page (`true`) or not (`false`).

**Signature**

```csharp
public Boolean fullWidthFeed {get; set;}
```

**Property Value**

*Type: Boolean*

**hideSidebar**

Specifies whether the sidebar is hidden (`true`) or not (`false`).

**Signature**

```csharp
public Boolean hideSidebar {get; set;}
```

**Property Value**

*Type: Boolean*
**highlightExternalFeedItems**
Controls whether to highlight external feed items (true) or not (false).

**Signature**
```
public Boolean highlightExternalFeedItems {get; set;}
```

**Property Value**
Type: Boolean

**leftComponents**
The individual components displayed in the left column of the feed view.

**Signature**
```
public List<Metadata.FeedLayoutComponent> leftComponents {get; set;}
```

**Property Value**
Type: List<FeedLayoutComponent Class>

**rightComponents**
Lists the individual components displayed in the right column of the feed view.

**Signature**
```
public List<Metadata.FeedLayoutComponent> rightComponents {get; set;}
```

**Property Value**
Type: List<FeedLayoutComponent Class>

**useInlineFiltersInConsole**
Indicates whether to use inline filters in the Salesforce console.

**Signature**
```
public Boolean useInlineFiltersInConsole {get; set;}
```

**Property Value**
Type: Boolean

**FeedLayout Methods**
The following are methods for FeedLayout.
clone()  
Makes a duplicate copy of the Metadata.FeedLayout.

Signature  
public Object clone()

Return Value  
Type: Object

FeedLayoutComponent Class

Represents a component in the feed view of a feed-based page layout.

Namespace

Metadata

Usage

Use this class when accessing Metadata.Layout metadata components. For more information, see “FeedLayoutComponent” in the Metadata API Developer Guide.

FeedLayoutComponent Properties

The following are properties for FeedLayoutComponent.

See FeedLayoutComponent in the Metadata API Developer Guide.

IN THIS SECTION:

componentType  
Represents a component in the feed view of a feed-based page layout. The type of component is required.

height  
The height, in pixels, of the component. Doesn’t apply to standardComponents.

page_x  
The name of the Visualforce page used as a custom component.
**componentType**
Represented as a component in the feed view of a feed-based page layout. The type of component is required.

**Signature**
```csharp
public Metadata.FeedLayoutComponentType componentType {get; set;}
```

**Property Value**
Type: Metadata.FeedLayoutComponentType on page 2400

**height**
The height, in pixels, of the component. Doesn’t apply to standardComponents

**Signature**
```csharp
public Integer height {get; set;}
```

**Property Value**
Type: Integer

**page_x**
The name of the Visualforce page used as a custom component.

**Signature**
```csharp
public String page_x {get; set;}
```

**Property Value**
Type: String

**FeedLayoutComponent Methods**
The following are methods for FeedLayoutComponent.

**IN THIS SECTION:**
```csharp
clone()
```
Makes a duplicate copy of the Metadata.FeedLayoutComponent.

```csharp
clone()
```
Makes a duplicate copy of the Metadata.FeedLayoutComponent.
Signature

public Object clone()

Return Value

Type: Object

FeedLayoutComponentType Enum

Indicates the type of feed layout component.

Enum Values

The following are the values of the Metadata.FeedLayoutComponentType enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CaseExperts</td>
<td>List of case experts.</td>
</tr>
<tr>
<td>CaseUnifiedFiles</td>
<td>List of all files attached to the case.</td>
</tr>
<tr>
<td>CustomButtons</td>
<td>Custom button.</td>
</tr>
<tr>
<td>CustomLinks</td>
<td>Custom link.</td>
</tr>
<tr>
<td>Followers</td>
<td>List of followers.</td>
</tr>
<tr>
<td>Following</td>
<td>Icon that toggles between a Follow button (if the user viewing a record doesn’t already follow it) and a Following indicator (if the user viewing a record does follow it).</td>
</tr>
<tr>
<td>HelpAndToolLinks</td>
<td>Icons that link to the help topic for the page, the page layout, and, the printable view of the page. Available only on Case layouts.</td>
</tr>
<tr>
<td>Milestones</td>
<td>Milestone tracker, which lets users see the status of a milestone on a case. Available only on Case layouts.</td>
</tr>
<tr>
<td>SimilarCases</td>
<td>List of similar cases.</td>
</tr>
<tr>
<td>Topics</td>
<td>List of topics related to the record.</td>
</tr>
<tr>
<td>Visualforce</td>
<td>Custom Visualforce component.</td>
</tr>
</tbody>
</table>

FeedLayoutFilter Class

Represents a feed filter option in the feed view of a feed-based page layout. A filter can have only standardFilter or feedItemType set.

Namespace

Metadata
Usage
Use this class when accessing `Metadata.Layout` metadata components. For more information, see “FeedLayoutFilter” in the `Metadata API Developer Guide`.

IN THIS SECTION:
  FeedLayoutFilter Properties
  FeedLayoutFilter Methods

FeedLayoutFilter Properties
The following are properties for FeedLayoutFilter.

IN THIS SECTION:
  feedFilterName
  The name of a CustomFeedFilter component. Names are prefixed with the name of the parent object. For example, `Case.MyCustomFeedFilter`.

  feedFilterType
  The type of filter.

  feedItemType
  The type of feed item to display.

**feedFilterName**
The name of a CustomFeedFilter component. Names are prefixed with the name of the parent object. For example, `Case.MyCustomFeedFilter`.

**Signature**
```
public String feedFilterName {get; set;}
```

**Property Value**
Type: `String`

**feedFilterType**
The type of filter.

**Signature**
```
public Metadata.FeedLayoutFilterType feedFilterType {get; set;}
```

**Property Value**
Type: `FeedLayoutFilterType Enum`
**feedItemType**
The type of feed item to display.

**Signature**
```
public Metadata.FeedItemTypeEnum feedItemType {get; set;}
```

**Property Value**
Type: `FeedItemTypeEnum` Enum

**FeedLayoutFilter Methods**
The following are methods for `FeedLayoutFilter`.

**IN THIS SECTION:**
```
clone()
```

Makes a duplicate copy of the `Metadata.FeedLayoutFilter`.

**clone()**
Makes a duplicate copy of the `Metadata.FeedLayoutFilter`.

**Signature**
```
public Object clone()
```

**Return Value**
Type: Object

**FeedLayoutFilterPosition Enum**
Describes where the feed filters list is included in the layout.

**Enum Values**
The following are the values of the `Metadata.FeedLayoutFilterPosition` enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CenterDropDown</td>
<td>As a drop-down list in the center column.</td>
</tr>
<tr>
<td>LeftFixed</td>
<td>As a fixed list in the left column.</td>
</tr>
<tr>
<td>LeftFloat</td>
<td>As a floating list in the left column.</td>
</tr>
</tbody>
</table>
FeedLayoutFilterType Enum

The type of feed layout filter.

Enum Values

The following are the values of the Metadata.FeedLayoutFilterType enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AllUpdates</td>
<td>Shows all feed items on a record.</td>
</tr>
<tr>
<td>Custom</td>
<td>Shows custom feed items.</td>
</tr>
<tr>
<td>FeedItemType</td>
<td>Shows feed items only for a particular type of activity on the record.</td>
</tr>
</tbody>
</table>

Layout Class

Represents the metadata associated with a page layout.

Namespace

Metadata

Usage

Use this class to access layout metadata components. For more information, see Layout in the Metadata API Developer Guide.

IN THIS SECTION:

- Layout Properties
- Layout Methods

Layout Properties

The following are properties for Layout.

IN THIS SECTION:

- customButtons
  - The custom buttons for this layout.
- customConsoleComponents
  - Represents custom console components (Visualforce pages, lookup fields, or related lists) on a page layout.
- emailDefault
  - Default value for the email checkbox. Only relevant if the showEmailCheckbox property is set.
- excludeButtons
  - List of standard buttons to exclude from this layout.
feedLayout
Represents the values that define the feed view of a feed-based page layout.

headers
Represents the layout headers used for tagging.

layoutSections
The main sections of the layout containing fields, s-controls, and custom links. The order here determines the layout order.

miniLayout
Represents a minilayout, which is used in the mini view of a record in the Console tab, hover details, and event overlays.

multilineLayoutFields
Fields for special multiline layout fields which appear in OpportunityProduct layouts.

platformActionList
The list of actions, and their order, that display in the Salesforce mobile action bar for the layout.

quickActionList
The list of quick actions that display in the full Salesforce site for the page layout.

relatedContent
The Related Content section of the page layout.

relatedLists
The related lists for the layout, listed in the order they appear in the user interface.

relatedObjects
The list of related objects that appears in the mini view of the console.

runAssignmentRulesDefault
Default value for the "run assignment rules" checkbox. Only relevant if the showRunAssignmentRulesCheckbox property is set.

showEmailCheckbox
Controls whether to show the email checkbox. Only allowed on Case, CaseClose, and Task layouts. The default state of checkbox is controlled by the emailDefault property.

showHighlightsPanel
If set, the highlights panel displays on pages in the Salesforce console.

showInteractionLogPanel
If set, the interaction log displays on pages in the Salesforce console.

showKnowledgeComponent
Only allowed on Case layouts. If set, the Knowledge sidebar displays on cases in the Salesforce console.

showRunAssignmentRulesCheckbox
Controls whether to show the Run Assignment Rules checkbox. Only allowed on Lead and Case layouts. The default state of checkbox is controlled by the runAssignmentRulesDefault property.

showSolutionSection
Only allowed on CaseClose layout. If set, the built-in solution information section shows up on the page.

showSubmitAndAttachButton
For Cast layouts only. If set, the Submit & Add Attachment button displays on case edit pages to portal users in the Customer Portal.

summaryLayout
The summary layout for this layout.
**customButtons**
The custom buttons for this layout.

**Signature**
public List<String> customButtons {get; set;}

**Property Value**
Type: List<String>

**customConsoleComponents**
Represents custom console components (Visualforce pages, lookup fields, or related lists) on a page layout.

**Signature**
public Metadata.CustomConsoleComponents customConsoleComponents {get; set;}

**Property Value**
Type: CustomConsoleComponents Class

**emailDefault**
Default value for the email checkbox. Only relevant if the showEmailCheckbox property is set.

**Signature**
public Boolean emailDefault {get; set;}

**Property Value**
Type: Boolean

**excludeButtons**
List of standard buttons to exclude from this layout.

**Signature**
public List<String> excludeButtons {get; set;}

**Property Value**
Type: List<String>

**feedLayout**
Represents the values that define the feed view of a feed-based page layout.
**Signature**

```java
public Metadata.FeedLayout feedLayout {get; set;}
```

**Property Value**

Type: `Metadata.FeedLayout`

**headers**

Represents the layout headers used for tagging.

**Signature**

```java
public List<Metadata.LayoutHeader> headers {get; set;}
```

**Property Value**

Type: `List<Metadata.LayoutHeader>`

**layoutSections**

The main sections of the layout containing fields, s-controls, and custom links. The order here determines the layout order.

**Signature**

```java
public List<Metadata.LayoutSection> layoutSections {get; set;}
```

**Property Value**

Type: `List<Metadata.LayoutSection>`

**miniLayout**

Represents a minilayout, which is used in the mini view of a record in the Console tab, hover details, and event overlays.

**Signature**

```java
public Metadata.MiniLayout miniLayout {get; set;}
```

**Property Value**

Type: `Metadata.MiniLayout`

**multilineLayoutFields**

Fields for special multiline layout fields which appear in OpportunityProduct layouts.

**Signature**

```java
public List<String> multilineLayoutFields {get; set;}
```
Property Value
Type: List<String>

platformActionList
The list of actions, and their order, that display in the Salesforce mobile action bar for the layout.

Signature
public Metadata.PlatformActionList platformActionList {get; set;}

Property Value
Type: Metadata.PlatformActionList

quickActionList
The list of quick actions that display in the full Salesforce site for the page layout.

Signature
public Metadata.QuickActionList quickActionList {get; set;}

Property Value
Type: Metadata.QuickActionList

relatedContent
The Related Content section of the page layout.

Signature
public Metadata.RelatedContent relatedContent {get; set;}

Property Value
Type: Metadata.RelatedContent

relatedLists
The related lists for the layout, listed in the order they appear in the user interface.

Signature
public List<Metadata.RelatedListItem> relatedLists {get; set;}

Property Value
Type: List<Metadata.RelatedListItem>
relatedObjects
The list of related objects that appears in the mini view of the console.

Signature
public List<String> relatedObjects {get; set;}

Property Value
Type: List<String>

runAssignmentRulesDefault
Default value for the “run assignment rules” checkbox. Only relevant if the showRunAssignmentRulesCheckbox property is set.

Signature
public Boolean runAssignmentRulesDefault {get; set;}

Property Value
Type: Boolean

showEmailCheckbox
Controls whether to show the email checkbox. Only allowed on Case, CaseClose, and Task layouts. The default state of checkbox is controlled by the emailDefault property.

Signature
public Boolean showEmailCheckbox {get; set;}

Property Value
Type: Boolean

showHighlightsPanel
If set, the highlights panel displays on pages in the Salesforce console.

Signature
public Boolean showHighlightsPanel {get; set;}

Property Value
Type: Boolean
showInteractionLogPanel
If set, the interaction log displays on pages in the Salesforce console.

Signature
public Boolean showInteractionLogPanel {get; set;}

Property Value
Type: Boolean

showKnowledgeComponent
Only allowed on Case layouts. If set, the Knowledge sidebar displays on cases in the Salesforce console.

Signature
public Boolean showKnowledgeComponent {get; set;}

Property Value
Type: Boolean

showRunAssignmentRulesCheckbox
Controls whether to show the Run Assignment Rules checkbox. Only allowed on Lead and Case layouts. The default state of checkbox is controlled by the runAssignmentRulesDefault property.

Signature
public Boolean showRunAssignmentRulesCheckbox {get; set;}

Property Value
Type: Boolean

showSolutionSection
Only allowed on CaseClose layout. If set, the built-in solution information section shows up on the page.

Signature
public Boolean showSolutionSection {get; set;}

Property Value
Type: Boolean
**showSubmitAndAttachButton**
For Cast layouts only. If set, the Submit & Add Attachment button displays on case edit pages to portal users in the Customer Portal.

**Signature**

```java
public Boolean showSubmitAndAttachButton {get; set;}
```

**Property Value**

Type: Boolean

**summaryLayout**
The summary layout for this layout.

**Signature**

```java
public Metadata.SummaryLayout summaryLayout {get; set;}
```

**Property Value**

Type: Metadata.SummaryLayout

### Layout Methods
The following are methods for `Layout`.

**IN THIS SECTION:**

- **clone()**
  Makes a duplicate copy of the `Metadata.Layout`.

- **clone()**
  Makes a duplicate copy of the `Metadata.Layout`.

  **Signature**

  ```java
  public Object clone()
  ```

  **Return Value**

  Type: Object

### LayoutColumn Class

Represents the items in a column within a layout section.
Namespace

Metadata

Usage

Use this class when accessing `Metadata.Layout` metadata components. For more information, see "LayoutColumn" in the `Metadata API Developer Guide`.

IN THIS SECTION:

- LayoutColumn Properties
- LayoutColumn Methods

LayoutColumn Properties

The following are properties for `LayoutColumn`.

IN THIS SECTION:

- layoutItems
- reserved

**layoutItems**

The individual items within a column (ordered from top to bottom).

**Signature**

```java
public List<Metadata.LayoutItem> layoutItems {get; set;}
```

**Property Value**

Type: `List<Metadata.LayoutItem>`

**reserved**

This field is reserved for Salesforce.

**Signature**

```java
public String reserved {get; set;}
```

**Property Value**

Type: `String`
**LayoutColumn Methods**

The following are methods for LayoutColumn.

**IN THIS SECTION:**

- `clone()`
  Makes a duplicate copy of the Metadata.LayoutColumn.

**clone()**

Makes a duplicate copy of the Metadata.LayoutColumn.

**Signature**

```java
public Object clone()
```

**Return Value**

Type: Object

**LayoutHeader Enum**

Represents tagging types used for Metadata.Layout.headers

**Enum Values**

The following are the values of the Metadata.LayoutHeader enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PersonalTagging</td>
<td>Tag is set to private user.</td>
</tr>
<tr>
<td>PublicTagging</td>
<td>Tag is viewable to any user who can access the record.</td>
</tr>
</tbody>
</table>

**LayoutItem Class**

Represents the valid values that define a layout item.

**Namespace**

Metadata

**Usage**

Use this class when accessing Metadata.Layout metadata components. For more information, see "LayoutItem" in the Metadata API Developer Guide.
IN THIS SECTION:

LayoutItem Properties
LayoutItem Methods

LayoutItem Properties

The following are properties for LayoutItem.

IN THIS SECTION:

analyticsCloudComponent
A Wave Analytics dashboard component on a page.

behavior
Determines the field behavior.

canvas
References a canvas app.

component
References a component.

customLink
The custom link reference.

emptySpace
Controls if this layout item is a blank space.

field
The field name reference, relative to the layout, for example “Description” or “MyField__c”.

height
For s-controls and pages only, the height in pixels.

page_x
Reference to a Visualforce page.

reportChartComponent
Refers to a report chart that you can add to a standard or custom object page.

scontrol
Reference to an s-control.

showLabel
For s-control and pages only, whether to show the label.

showScrollbars
For s-control and pages only, whether to show scrollbars.

width
For s-control and pages only, the width in pixels or percent. Pixel values are simply the number of pixels, for example, 500. Percentage values must include the percent sign, for example, 20%.

analyticsCloudComponent
A Wave Analytics dashboard component on a page.
**Signature**

```csharp
public Metadata.AnalyticsCloudComponentLayoutItem analyticsCloudComponent {get; set;}
```

**Property Value**

Type: `Metadata.AnalyticsCloudComponentLayoutItem`

---

**behavior**

Determines the field behavior.

**Signature**

```csharp
public Metadata.UiBehavior behavior {get; set;}
```

**Property Value**

Type: `Metadata.UiBehavior`

---

**canvas**

References a canvas app.

**Signature**

```csharp
public String canvas {get; set;}
```

**Property Value**

Type: `String`

---

**component**

References a component.

**Signature**

```csharp
public String component {get; set;}
```

**Property Value**

Type: `String`

---

**customLink**

The custom link reference.

**Signature**

```csharp
public String customLink {get; set;}
```
Property Value
Type: String

**emptySpace**
Controls if this layout item is a blank space.

Signature
```
public Boolean emptySpace {get; set;}
```

Property Value
Type: Boolean

**field**
The field name reference, relative to the layout, for example "Description" or "MyField__c".

Signature
```
public String field {get; set;}
```

Property Value
Type: String

**height**
For s-controls and pages only, the height in pixels.

Signature
```
public Integer height {get; set;}
```

Property Value
Type: Integer

**page_x**
Reference to a Visualforce page.

Signature
```
public String page_x {get; set;}
```

Property Value
Type: String
**reportChartComponent**
Refers to a report chart that you can add to a standard or custom object page.

**Signature**
```csharp
public Metadata.ReportChartComponentLayoutItem reportChartComponent {get; set;}
```

**Property Value**
Type: `Metadata.ReportChartComponentLayoutItem`

**scontrol**
Reference to an s-control.

**Signature**
```csharp
public String scontrol {get; set;}
```

**Property Value**
Type: `String`

**showLabel**
For s-control and pages only, whether to show the label.

**Signature**
```csharp
public Boolean showLabel {get; set;}
```

**Property Value**
Type: `Boolean`

**showScrollbars**
For s-control and pages only, whether to show scrollbars.

**Signature**
```csharp
public Boolean showScrollbars {get; set;}
```

**Property Value**
Type: `Boolean`
**width**
For s-control and pages only, the width in pixels or percent. Pixel values are simply the number of pixels, for example, 500. Percentage values must include the percent sign, for example, 20%.

**Signature**
```java
public String width {get; set;}
```

**Property Value**
Type: String

**LayoutItem Methods**
The following are methods for LayoutItem.

**In This Section:**
- `clone()`  
  Makes a duplicate copy of the Metadata.LayoutItem.

**clone()**
Makes a duplicate copy of the Metadata.LayoutItem.

**Signature**
```java
public Object clone()
```

**Return Value**
Type: Object

**LayoutSection Class**
Represents a section of a page layout, such as the Custom Links section.

**Namespace**
Metadata

**Usage**
Use this class when accessing Metadata.Layout metadata components. For more information, see “LayoutSection” in the Metadata API Developer Guide.

**In This Section:**
- LayoutSection Properties
## LayoutSection Properties

The following are properties for `LayoutSection`.

### IN THIS SECTION:

- **customLabel**
  Indicates if this section’s label is custom or standard (built-in).

- **detailHeading**
  Controls if this section appears in the detail page.

- **editHeading**
  Controls if this section appears in the edit page.

- **label**
  The label; either standard or custom, based on the `customLabel` property.

- **layoutColumns**
  Lists the layout columns. You can have one, two, or three columns, ordered left to right, are possible.

- **style**
  The style of the layout for this section.

---

### customLabel

Indicates if this section's label is custom or standard (built-in).

**Signature**

```java
public Boolean customLabel {get; set;}
```

**Property Value**

Type: `Boolean`

---

### detailHeading

Controls if this section appears in the detail page.

**Signature**

```java
public Boolean detailHeading {get; set;}
```

**Property Value**

Type: `Boolean`

---

### editHeading

Controls if this section appears in the edit page.

**Signature**

```java
public Boolean editHeading {get; set;}
```

**Property Value**

Type: `Boolean`
Signature
public Boolean editHeading {get; set;}

Property Value
Type: Boolean

label
The label; either standard or custom, based on the customLabel property.

Signature
public String label {get; set;}

Property Value
Type: String

layoutColumns
Lists the layout columns. You can have one, two, or three columns, ordered left to right, are possible.

Signature
public List<Metadata.LayoutColumn> layoutColumns {get; set;}

Property Value
Type: List<Metadata.LayoutColumn>

style
The style of the layout for this section.

Signature
public Metadata.LayoutSectionStyle style {get; set;}

Property Value
Type: Metadata.LayoutSectionStyle

LayoutSection Methods
The following are methods for LayoutSection.
IN THIS SECTION:

**clone()**

Makes a duplicate copy of the `Metadata.LayoutSection`.

**Signature**

```java
public Object clone()
```

**Return Value**

Type: Object

---

**LayoutSectionStyle Enum**

Describes the possible styles for a layout section.

**Enum Values**

The following are the values of the `Metadata.LayoutSectionStyle` enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CustomLinks</td>
<td>Contains custom links only</td>
</tr>
<tr>
<td>OneColumn</td>
<td>One column</td>
</tr>
<tr>
<td>TwoColumnsLeftToRight</td>
<td>Two columns, tab goes left to right</td>
</tr>
<tr>
<td>TwoColumnsTopToBottom</td>
<td>Two columns, tab goes top to bottom</td>
</tr>
</tbody>
</table>

---

**Metadata Class**

An abstract base class that represents a custom metadata component.

**Namespace**

`Metadata`

**Usage**

You can't create instances of this abstract class. Instead, create an instance of a specific custom metadata component class that derives from `Metadata`, such as `Metadata.CustomMetadata`. For more information, see `Metadata` in the `Metadata API Developer Guide`.
IN THIS SECTION:
  Metadata Properties
  Metadata Methods

Metadata Properties
The following are properties for Metadata.

IN THIS SECTION:
  fullName
  The full name of the custom metadata, which can include the namespace, type, and component name.

**fullName**
The full name of the custom metadata, which can include the namespace, type, and component name.

**Signature**
```java
public String fullName {get; set;}
```

**Property Value**
*Type: String*
The format of the full name can include the namespace, metadata type, and metadata component name. If you’re updating components in a namespace, you also need to qualify the namespace for the component in the full name. For example, the full name for a custom metadata "MDType1__mdt" component named "Component1" that is contained in the "myPackage" namespace is "myPackage__MDType1__mdt.myPackage__Component1". For more information on full name formats for different metadata types, see reference documentation on the metadata types in the Metadata API Developer Guide.

Metadata Methods
The following are methods for Metadata.

IN THIS SECTION:
  clone()
  Makes a duplicate copy of the Metadata.Metadata.

**clone()**
Makes a duplicate copy of the Metadata.Metadata.

**Signature**
```java
public Object clone()
```
**Return Value**

Type: Object

---

**MetadataType Enum**

Represents the custom metadata components available in Apex.

**Enum Values**

The following are the values of the `Metadata.MetadataType` enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CustomMetadata</td>
<td>Records of custom metadata types</td>
</tr>
<tr>
<td>Layout</td>
<td>Layouts</td>
</tr>
</tbody>
</table>

**MetadataValue Class**

An abstract base class that represents a custom metadata component field.

**Namespace**

`Metadata`

**Usage**

You can't create instances of this abstract class. Instead, create an instance of a specific custom metadata component value class that derives from `Metadata.MetadataValue`, such as `Metadata.CustomMetadataValue`.

IN THIS SECTION:

`MetadataValue Methods`

**MetadataValue Methods**

The following are methods for `MetadataValue`.

IN THIS SECTION:

`clone()`

Makes a duplicate copy of the `Metadata.MetadataValue`.

`clone()`

Makes a duplicate copy of the `Metadata.MetadataValue`. 
MiniLayout Class

Represents a mini view of a record in the Console tab, hover details, and event overlays.

Namespace

Metadata

Usage

Use this class when accessing Metadata.Layout metadata components. For more information, see “MiniLayout” in the Metadata API Developer Guide.

IN THIS SECTION:

MiniLayout Properties
MiniLayout Methods

MiniLayout Properties

The following are properties for MiniLayout.

IN THIS SECTION:

fields
The fields for the mini-layout, listed in the order they appear in the UI. Fields that appear in the mini-layout must appear in the main layout.

relatedLists
The mini related lists, listed in the order they appear in the UI. You cannot set sorting on mini related lists. Fields that appear in the mini related lists must appear in the main layout.

fields
The fields for the mini-layout, listed in the order they appear in the UI. Fields that appear in the mini-layout must appear in the main layout.

Signature

public List<String> fields {get; set;}

Return Value

Type: Object
**relatedLists**

The mini related lists, listed in the order they appear in the UI. You cannot set sorting on mini related lists. Fields that appear in the mini related lists must appear in the main layout.

**Signature**

```java
public List<Metadata.RelatedListItem> relatedLists {get; set;}
```

**MiniLayout Methods**

The following are methods for `MiniLayout`.

**IN THIS SECTION:**

`clone()`

Makes a duplicate copy of the `Metadata.MiniLayout`.

**clone()**

Makes a duplicate copy of the `Metadata.MiniLayout`.

**Signature**

```java
public Object clone()
```

**Return Value**

Type: Object

**Operations Class**

Represents a class to execute metadata operations, such as retrieving or deploying custom metadata.

**Namespace**

`Metadata`

**Usage**

Use the `Metadata.Operations` class to execute metadata operations. For more information on use cases and restrictions of metadata operations in Apex, see `Metadata`. 
Example: Retrieve Metadata

The following example retrieves the "MyTestCustomMDType" custom metadata record from the subscriber org, and inspects the custom fields.

```java
public class ReadMetadata {
    public void retrieveMetadata () {
        // List fullnames of components we want to retrieve
        List<String> componentNameList = new List<String>{'ISVNamespace__TestCustomMDType.MyTestCustomMDType'};

        // Retrieve components that are records of custom metadata types
        // based on name
        List<Metadata.Metadata> components = Metadata.Operations.retrieve(Metadata.MetadataType.CustomMetadata, componentNameList);
        Metadata.CustomMetadata customMetadataRecord = (Metadata.CustomMetadata) components.get(0);

        // Check fields of retrieved component
        List<Metadata.CustomMetadataValue> values = customMetadataRecord.values;
        for (integer i = 0; i < values.size(); i++) {
            if (values.get(i).field == 'testField__c' && values.get(i).value == 'desired value') {
                // ...process accordingly...
            }
        }
    }
}
```

Example: Deploy Metadata

The following example uses the Metadata API in Apex to update the customField custom field value of the MetadataRecordName custom metadata record and deploy this change into the subscriber org. Because the deployment is asynchronous, you must provide a callback class that implements the Metadata.DeployCallback interface, which is then used when the queued deployment completes.

```java
public class CreateMetadata {
    public void updateAndDeployMetadata() {
        // Setup custom metadata to be created in the subscriber org.
        Metadata.CustomMetadata customMetadata = new Metadata.CustomMetadata();
        customMetadata.fullName = 'ISVNamespace__MetadataTypeName.MetadataRecordName';

        Metadata.CustomMetadataValue customField = new Metadata.CustomMetadataValue();
        customField.field = 'customField__c';
        customField.value = 'New value';

        customMetadata.values.add(customField);

        Metadata.DeployContainer mdContainer = new Metadata.DeployContainer();
        mdContainer.addMetadata(customMetadata);

        // Setup deploy callback, MyDeployCallback implements
        // the Metadata.DeployCallback interface (code for
        // this class not shown in this example)
        MyDeployCallback callback = new MyDeployCallback();
    }
}
```
// Enqueue custom metadata deployment
Id jobId = Metadata.Operations.enqueueDeployment(mdContainer, callback);
}

Example: Create Two Metadata Records Synchronously
Create a metadata record along with another one that references it in the same transaction. If the parent record was installed with a namespace, prefix the developer name with `recordNs__`.

Note: No custom metadata relationship can relate records of the same type to each other.

```apex
public class CreateMetadata {
    public Id doCreate(
        String parentRecDevName,
        String parentRecLabel,
        String childRecDevName,
        String childRecLabel) {

        Metadata.DeployContainer mdContainer = new Metadata.DeployContainer();

        Metadata.CustomMetadata parentRecord = new Metadata.CustomMetadata();
        parentRecord.fullName = 'ParentType.' + parentRecDevName;
        parentRecord.label = parentRecLabel;
        mdContainer.addMetadata(parentRecord);

        Metadata.CustomMetadata childRecord = new Metadata.CustomMetadata();
        childRecord.fullName = 'ChildType.' + childRecDevName;
        childRecord.label = childRecLabel;
        Metadata.CustomMetadataValue relValue = new Metadata.CustomMetadataValue();
        relValue.field = 'Parent__c';
        relValue.value = parentRecDevName;
        childRecord.values.add(relValue);
        mdContainer.addMetadata(childRecord);

        Id jobId = Metadata.Operations.enqueueDeployment(mdContainer, null);
        return jobId;
    }
}
```

IN THIS SECTION:
- Operations Methods

Operations Methods
The following are methods for Operations.
IN THIS SECTION:

- **clone()**

- **enqueueDeployment(container, callback)**
  Deploys custom metadata components asynchronously.

- **retrieve(type, fullNames)**
  Retrieves a list of custom metadata components.

---

**clone()**


**Signature**

```java
public Object clone()
```

**Return Value**

Type: Object

---

**enqueueDeployment(container, callback)**

Deploys custom metadata components asynchronously.

**Signature**

```java
public static Id enqueueDeployment(Metadata.DeployContainer container,
                                   Metadata.DeployCallback callback)
```

**Parameters**

- **container**
  Type: Metadata.DeployContainer
  Container that contains the set of metadata components to deploy.

- **callback**
  Type: Metadata.DeployCallback
  A class that implements the Metadata.DeployCallback interface. Used by Salesforce to return information about the deployment results.

**Return Value**

Type: Id

ID of deployment request.

---

**retrieve(type, fullNames)**

Retrieves a list of custom metadata components.
Signature

```java
public static List<Metadata.Metadata> retrieve(Metadata.MetadataType type, List<String> fullNames)
```

Parameters

type
Type: Metadata.MetadataType
The metadata component type.

fullNames
Type: List<String>
A list of component names to retrieve. For information on component name formats, see Metadata.fullName().

Return Value
Type: List<Metadata.Metadata>

PlatformActionList Class

Represents the list of actions, and their order, that display in the Salesforce mobile action bar for the layout.

Namespace
Metadata

Usage
Use this class when accessing Metadata.Layout metadata components. For more information, see “PlatformActionList” in the Metadata API Developer Guide.

IN THIS SECTION:
  PlatformActionList Properties
  PlatformActionList Methods

PlatformActionList Properties

The following are properties for PlatformActionList.

IN THIS SECTION:
  actionListContext
  The context of the action list.
  platformActionListItems
  The actions in the platform action list.
relatedSourceEntity
When the actionListContext property is "RelatedList" or "RelatedListRecord", this field represents the API name of the related list to which the action belongs.

actionListContext
The context of the action list.

Signature
public Metadata.PlatformActionListContextEnum actionListContext {get; set;}

Property Value
Type: Metadata.PlatformActionListContextEnum

platformActionListItems
The actions in the platform action list.

Signature
public List<Metadata.PlatformActionListItem> platformActionListItems {get; set;}

Property Value
Type: List<Metadata.PlatformActionListItem>

relatedSourceEntity
When the actionListContext property is "RelatedList" or "RelatedListRecord", this field represents the API name of the related list to which the action belongs.

Signature
public String relatedSourceEntity {get; set;}

Property Value
Type: String

PlatformActionList Methods
The following are methods for PlatformActionList.

IN THIS SECTION:
    clone()
    Makes a duplicate copy of the Metadata.PlatformActionList.
clone()

Makes a duplicate copy of the Metadata.PlatformActionList.

Signature

public Object clone()

Return Value

Type: Object

PlatformActionListContextEnum Enum

Describes the different contexts of action lists.

Enum Values

The following are the values of the Metadata.PlatformActionListContextEnum enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ActionDefinition</td>
<td>Action definition context.</td>
</tr>
<tr>
<td>Assistant</td>
<td>Assistant context.</td>
</tr>
<tr>
<td>BannerPhoto</td>
<td>Banner photo context.</td>
</tr>
<tr>
<td>Chatter</td>
<td>Chatter context.</td>
</tr>
<tr>
<td>Dockable</td>
<td>Dockable context.</td>
</tr>
<tr>
<td>FeedElement</td>
<td>Feed element context.</td>
</tr>
<tr>
<td>Flexipage</td>
<td>Flexipage context.</td>
</tr>
<tr>
<td>Global_x</td>
<td>Global context.</td>
</tr>
<tr>
<td>ListView</td>
<td>Listview context.</td>
</tr>
<tr>
<td>ListViewDefinition</td>
<td>Listview definition context.</td>
</tr>
<tr>
<td>ListViewRecord</td>
<td>Listview record context.</td>
</tr>
<tr>
<td>Lookup</td>
<td>Lookup context.</td>
</tr>
<tr>
<td>MruList</td>
<td>MRU list context.</td>
</tr>
<tr>
<td>MruRow</td>
<td>MRU row context.</td>
</tr>
<tr>
<td>ObjectHomeChart</td>
<td>Object home chart context.</td>
</tr>
<tr>
<td>Photo</td>
<td>Photo context.</td>
</tr>
<tr>
<td>Record</td>
<td>Record context.</td>
</tr>
<tr>
<td>RecordEdit</td>
<td>Record edit context</td>
</tr>
</tbody>
</table>
Related list context.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RelatedList</td>
<td>Related list context.</td>
</tr>
<tr>
<td>RelatedListRecord</td>
<td>Related list record context.</td>
</tr>
</tbody>
</table>

### PlatformActionListItem Class

Represents an action in the platform action list for a layout.

#### Namespace

**Metadata**

#### Usage

Use this class when accessing `Metadata.Layout` metadata components. For more information, see “PlatformActionListItem” in the Metadata API Developer Guide.

#### PlatformActionListItem Properties

The following are properties for `PlatformActionListItem`.

#### actionName

The API name for the action in the list.

#### actionType

The type of action.

#### sortOrder

The placement of the action in the list.

#### subtype

The subtype of the action.

### actionName

The API name for the action in the list.

### Signature

```java
public String actionName {get; set;}
```
Property Value
Type: String

**actionType**
The type of action.

Signature
`public Metadata.PlatformActionTypeEnum actionType {get; set;}`

Property Value
Type: `Metadata.PlatformActionTypeEnum`

**sortOrder**
The placement of the action in the list.

Signature
`public Integer sortOrder {get; set;}`

Property Value
Type: `Integer`

**subtype**
The subtype of the action.

Signature
`public String subtype {get; set;}`

Property Value
Type: `String`

**PlatformActionListItem Methods**
The following are methods for `PlatformActionListItem`.

IN THIS SECTION:
- `clone()`
  Makes a duplicate copy of the `Metadata.PlatformActionListItem`. 
clone()

Makes a duplicate copy of the Metadata.PlatformActionListItem.

Signature

public Object clone()

Return Value

Type: Object

PlatformActionTypeEnum Enum

The type of action for a PlatformActionListItem.

Enum Values

The following are the values of the Metadata.PlatformActionTypeEnum enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ActionLink</td>
<td>An indicator on a feed element that targets an API, a web page, or a file, represented by a button in the Salesforce Chatter feed UI.</td>
</tr>
<tr>
<td>CustomButton</td>
<td>When clicked, opens a URL or a Visualforce page in a window or executes JavaScript.</td>
</tr>
<tr>
<td>InvocableAction</td>
<td>An invocable action such as posting to Chatter.</td>
</tr>
<tr>
<td>ProductivityAction</td>
<td>Productivity actions are predefined by Salesforce and are attached to a limited set of objects. You can’t edit or delete productivity actions.</td>
</tr>
<tr>
<td>QuickAction</td>
<td>A global or object-specific action.</td>
</tr>
<tr>
<td>StandardButton</td>
<td>A predefined Salesforce button such as New, Edit, and Delete.</td>
</tr>
</tbody>
</table>

PrimaryTabComponents Class

Represents custom console components on primary tabs in the Salesforce console.

Namespace

Metadata

Usage

Use this class when accessing Metadata.Layout metadata components. For more information, see “PrimaryTabComponents” in the Metadata API Developer Guide.
**PrimaryTabComponents Properties**

The following are properties for `PrimaryTabComponents`.

**component**

Represents a custom console component (Visualforce page, lookup field, or related lists) on a section of a page layout.

**containers**

Represents a location and style in which to display more than one custom console component on the sidebars of the Salesforce console.

**Signature**

```java
public List<Metadata.ConsoleComponent> component {get; set;}
```

**Property Value**

Type: `List<Metadata.ConsoleComponent>`

**containers**

Represents a location and style in which to display more than one custom console component on the sidebars of the Salesforce console.

**Signature**

```java
public List<Metadata.Container> containers {get; set;}
```

**Property Value**

Type: `List<Metadata.Container>`

**PrimaryTabComponents Methods**

The following are methods for `PrimaryTabComponents`.

**clone()**

Makes a duplicate copy of the `Metadata.PrimaryTabComponents`.
clone()

Makes a duplicate copy of the Metadata.PrimaryTabComponents.

Signature

public Object clone()

Return Value

Type: Object

QuickActionList Class

Represents the list of actions associated with the page layout.

Namespace

Metadata

Usage

Use this class when accessing Metadata.Layout metadata components. For more information, see “QuickActionList” in the Metadata API Developer Guide.

IN THIS SECTION:

QuickActionList Properties
QuickActionList Methods

QuickActionList Properties

The following are properties for QuickActionList.

IN THIS SECTION:

quickActionListItems

List of QuickActionList objects.

quickActionListItems

List of QuickActionList objects.

Signature

public List<Metadata.QuickActionListItem> quickActionListItems {get; set;}

Property Value

Type: List<Metadata.QuickActionListItem>
QuickActionList Methods

The following are methods for QuickActionList.

IN THIS SECTION:

  clone()
  Makes a duplicate copy of the Metadata.QuickActionList.

clone()

Makes a duplicate copy of the Metadata.QuickActionList.

Signature

public Object clone()

Return Value

Type: Object

QuickActionListItem Class

Represents an action in the QuickActionList.

Namespace

Metadata

Usage

Use this class when accessing Metadata.Layout metadata components. For more information, see “QuickActionListItem” in the Metadata API Developer Guide.

QuickActionListItem Properties

The following are properties for QuickActionListItem.

IN THIS SECTION:

quickActionName
The API name of the action.
**quickActionName**
The API name of the action.

**Signature**
```java
public String quickActionName {get; set;}
```

**Property Value**
Type: String

**QuickActionListItem Methods**
The following are methods for `QuickActionListItem`.

IN THIS SECTION:
- `clone()`

  Makes a duplicate copy of the `Metadata.QuickActionListItem`.

**clone()**
Makes a duplicate copy of the `Metadata.QuickActionListItem`.

**Signature**
```java
public Object clone()
```

**Return Value**
Type: Object

**RelatedContent Class**
Represents the Mobile Cards section of the page layout.

**Namespace**
`Metadata`

**Usage**
Use this class when accessing `Metadata.Layout` metadata components. For more information, see “RelatedContent” in the `Metadata API Developer Guide`.

IN THIS SECTION:
- RelatedContent Properties
- RelatedContent Methods
RelatedContent Properties
The following are properties for RelatedContent.

IN THIS SECTION:

**relatedContentItems**
A list of layout items in the Mobile Cards section of the page layout.

**Signature**
```
public List<Metadata.RelatedContentItem> relatedContentItems {get; set;}
```

**Property Value**
Type: `List<Metadata.RelatedContentItem>`

RelatedContent Methods
The following are methods for RelatedContent.

IN THIS SECTION:

**clone()**
Makes a duplicate copy of the `Metadata.RelatedContent`.

**Signature**
```
public Object clone()
```

**Return Value**
Type: `Object`

RelatedContentItem Class
Represents an individual item in the RelatedContent list.

**Namespace**
`Metadata`
Usage
Use this class when accessing `Metadata.Layout` metadata components. For more information, see “RelatedContentItem” in the `Metadata API Developer Guide`.

IN THIS SECTION:
- RelatedContentItem Properties
- RelatedContentItem Methods

RelatedContentItem Properties
The following are properties for `RelatedContentItem`.

IN THIS SECTION:
- layoutItem
  An individual layout item in the Mobile Cards section.

**layoutItem**
An individual layout item in the Mobile Cards section.

**Signature**
```csharp
public Metadata.LayoutItem layoutItem {get; set;}
```

**Property Value**
Type: `Metadata.LayoutItem`

RelatedContentItem Methods
The following are methods for `RelatedContentItem`.

IN THIS SECTION:
- clone()
  Makes a duplicate copy of the `Metadata.RelatedContentItem`.

**clone()**
Makes a duplicate copy of the `Metadata.RelatedContentItem`.

**Signature**
```csharp
public Object clone()
```
RelatedList Class

Represents related list custom components on the sidebars of the Salesforce console.

Namespace

Metadata

Usage

Use this class when accessing Metadata.Layout metadata components. For more information, see “RelatedList” in the Metadata API Developer Guide.

IN THIS SECTION:

RelatedList Properties
RelatedList Methods

RelatedList Properties

The following are properties for RelatedList.

IN THIS SECTION:

hideOnDetail
When set to true, the related list is hidden from detail pages where it appears as a component to prevent duplicate information from showing.

name
The name of the component as it appears to console users.

hideOnDetail

When set to true, the related list is hidden from detail pages where it appears as a component to prevent duplicate information from showing.

Signature

public Boolean hideOnDetail {get; set;}

Property Value

Type: Boolean
name

The name of the component as it appears to console users.

Signature

```java
public String name {get; set;}
```

Property Value

Type: String

RelatedList Methods

The following are methods for RelatedList.

IN THIS SECTION:

- clone()

  Makes a duplicate copy of the Metadata.RelatedList.

clone()

Makes a duplicate copy of the Metadata.RelatedList.

Signature

```java
public Object clone()
```

Return Value

Type: Object

RelatedListItem Class

Represents an item in the related list in a page layout.

Namespace

Metadata

Usage

Use this class when accessing Metadata.Layout metadata components. For more information, see “RelatedListItem” in the Metadata API Developer Guide.

IN THIS SECTION:

- RelatedListItem Properties
- RelatedListItem Methods
RelatedListItem Properties

The following are properties for RelatedListItem.

IN THIS SECTION:
  customButtons
  A list of custom buttons used in the related list.
  excludeButtons
  A list of excluded related-list buttons.
  fields
  A list of fields displayed in the related list. Uses aliases instead of field or API names.
  relatedList
  The name of the related list.
  sortField
  The name of the field used for sorting.
  sortOrder
  When sortField is set, the sortOrder property determines the sort order.

customButtons

A list of custom buttons used in the related list.

Signature

public List<String> customButtons {get; set;}

Property Value

Type: List<String>

For more information, see “Define Custom Buttons and Links” in the Salesforce online help.

excludeButtons

A list of excluded related-list buttons.

Signature

public List<String> excludeButtons {get; set;}

Property Value

Type: List<String>

fields

A list of fields displayed in the related list. Uses aliases instead of field or API names.
Signatures

**fields**

*public List<String> fields {get; set;}*

**relatedList**

The name of the related list.

*Signature*

*public String relatedList {get; set;)*

**sortField**

The name of the field used for sorting.

*Signature*

*public String sortField {get; set;)*

**sortOrder**

When `sortField` is set, the `sortOrder` property determines the sort order.

*Signature*

*public Metadata.SortOrder sortOrder {get; set;)*

**RelatedListItem Methods**

The following are methods for `RelatedListItem`.
IN THIS SECTION:

**clone()**

Makes a duplicate copy of the Metadata.RelatedListItem.

**Signature**

```java
public Object clone()
```

**Return Value**

Type: Object

---

**ReportChartComponentLayoutItem Class**

Represents the settings for a report chart on a standard or custom page.

**Namespace**

Metadata

**Usage**

Use this class when accessing Metadata.Layout metadata components. For more information, see “ReportChartComponentLayoutItem” in the Metadata API Developer Guide.

---

**ReportChartComponentLayoutItem Properties**

The following are properties for ReportChartComponentLayoutItem.

**IN THIS SECTION:**

- ReportChartComponentLayoutItem Properties
- ReportChartComponentLayoutItem Methods

**cacheData**

Indicates whether to use cached data when displaying the chart. When the attribute is set to true, data is cached for 24 hours. When the attribute is set to false, the report is run every time the page is refreshed.

**contextFilterableField**

Unique development name of the field by which a report chart is filtered to return data relevant to the page. If set, the ID field for the parent object of the page or report type is the chart data filter. The parent object for the report type and the page must match for a chart to return relevant data.
error
Error string that is populated only when an error occurs in the underlying report.

hideOnError
Controls whether users see a chart that has an error. When an error occurs and this attribute is not set, the chart doesn’t show any data except the error. Set the attribute to true to hide the chart from a page on error.

includeContext
If true, filters the report chart to return data that’s relevant to the page.

reportName
Unique development name of a report that includes a chart.

showTitle
If true, applies the title from the report to the chart.

size
Size of the displayed chart. The default is medium.

cacheData
Indicates whether to use cached data when displaying the chart. When the attribute is set to true, data is cached for 24 hours. When the attribute is set to false, the report is run every time the page is refreshed.

Signature
public Boolean cacheData {get; set;}

Property Value
Type: Boolean

contextFilterableField
Unique development name of the field by which a report chart is filtered to return data relevant to the page. If set, the ID field for the parent object of the page or report type is the chart data filter. The parent object for the report type and the page must match for a chart to return relevant data.

Signature
public String contextFilterableField {get; set;}

Property Value
Type: String

error
Error string that is populated only when an error occurs in the underlying report.

Signature
public String error {get; set;}

2445
Property Value
Type: String

hideOnError
Controls whether users see a chart that has an error. When an error occurs and this attribute is not set, the chart doesn’t show any data except the error. Set the attribute to true to hide the chart from a page on error.

Signature
public Boolean hideOnError {get; set;}

Property Value
Type: Boolean

includeContext
If true, filters the report chart to return data that’s relevant to the page.

Signature
public Boolean includeContext {get; set;}

Property Value
Type: Boolean

reportName
Unique development name of a report that includes a chart.

Signature
public String reportName {get; set;}

Property Value
Type: String

showTitle
If true, applies the title from the report to the chart.

Signature
public Boolean showTitle {get; set;}
Property Value
Type: Boolean

size
Size of the displayed chart. The default is medium.

Signature
public Metadata.ReportChartComponentSize size {get; set;}

Property Value
Type: Metadata.ReportChartComponentSize

ReportChartComponentLayoutItem Methods
The following are methods for ReportChartComponentLayoutItem.

IN THIS SECTION:
clone()  

clone ()

Signature
public Object clone()

Return Value
Type: Object

ReportChartComponentSize Enum
Describes the size of the displayed report chart component.

Enum Values
The following are the values of the Metadata.ReportChartComponentSize enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LARGE</td>
<td>Large chart size.</td>
</tr>
<tr>
<td>MEDIUM</td>
<td>Medium chart size.</td>
</tr>
</tbody>
</table>
SidebarComponent Class

Represents a specific custom console component to display in a container that hosts multiple components in one of the sidebars of the Salesforce console.

XML

NAMESPACE

Usage

Use this class when accessing Metadata.Layout metadata components. For more information, see “SidebarComponent” in the Metadata API Developer Guide.

IN THIS SECTION:

SidebarComponent Properties

SidebarComponent Methods

SidebarComponent Properties

The following are properties for SidebarComponent.

IN THIS SECTION:

componentType

Specifies the component type. Valid values are “KnowledgeOne”, “Lookup”, “Milestones”, “RelatedList”, “Topics”, “Files”, and “CaseExperts”.

createAction

If the component is a lookup field, the name of the quick action used to create a record.

enableLinking

If the component is a lookup field, lets users associate a record with this field.

height

The height of the component in the container. The unit property determines the unit of measurement, in pixels or percent.

knowledgeOneEnable

Indicates if the component is enabled for Knowledge One.

label

The name of the component as it displays to console users. Available for components in a container with the style of tabs or accordion.

lookup

If the component is a lookup field, the name of the field.
page_x
If the component is a Visualforce page, the name of the Visualforce page.

relatedLists
If the component is a related list component, the list of related list names.

unit
The unit of measurement (pixels or percent) for the height and width of the component in the container.

updateAction
If the component is a lookup field, the name of the quick action used to update a record.

width
The width of the component in the container. The unit property determines the unit of measurement, in pixels or percent.

**componentType**
Specifies the component type. Valid values are “KnowledgeOne”, “Lookup”, “Milestones”, “RelatedList”, “Topics”, “Files”, and “CaseExperts”.

**Signature**

```java
public String componentType {get; set;}
```

**Property Value**
Type: String

**createAction**
If the component is a lookup field, the name of the quick action used to create a record.

**Signature**

```java
public String createAction {get; set;}
```

**Property Value**
Type: String

**enableLinking**
If the component is a lookup field, lets users associate a record with this field.

**Signature**

```java
public Boolean enableLinking {get; set;}
```

**Property Value**
Type: Boolean
**height**
The height of the component in the container. The `unit` property determines the unit of measurement, in pixels or percent.

**Signature**
```java
public Integer height {get; set;}
```

**Property Value**
Type: Integer

**knowledgeOneEnable**
Indicates if the component is enabled for Knowledge One.

**Signature**
```java
public Boolean knowledgeOneEnable {get; set;}
```

**Property Value**
Type: Boolean

**label**
The name of the component as it displays to console users. Available for components in a container with the style of tabs or accordion.

**Signature**
```java
public String label {get; set;}
```

**Property Value**
Type: String

**lookup**
If the component is a lookup field, the name of the field.

**Signature**
```java
public String lookup {get; set;}
```

**Property Value**
Type: String

**page_x**
If the component is a Visualforce page, the name of the Visualforce page.
Signature

public String page_x {get; set;}

Property Value
Type: String

relatedLists
If the component is a related list component, the list of related list names.

Signature

public List<Metadata.RelatedList> relatedLists {get; set;}

Property Value
Type: List<Metadata.RelatedList>

unit
The unit of measurement (pixels or percent) for the height and width of the component in the container.

Signature

public String unit {get; set;}

Property Value
Type: String

updateAction
If the component is a lookup field, the name of the quick action used to update a record.

Signature

public String updateAction {get; set;}

Property Value
Type: String

width
The width of the component in the container. The unit property determines the unit of measurement, in pixels or percent.

Signature

public Integer width {get; set;}
Property Value
Type: Integer

SidebarComponent Methods
The following are methods forSidebarComponent.

IN THIS SECTION:

**clone()**
Makes a duplicate copy of the Metadata.SidebarComponent.

**Signature**
```
public Object clone()
```

**Return Value**
Type: Object

SortOrder Enum
Describes the sort order of a related list.

**Enum Values**
The following are the values of the Metadata.SortOrder enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asc_x</td>
<td>Sort in ascending order.</td>
</tr>
<tr>
<td>Desc_x</td>
<td>Sort in descending order.</td>
</tr>
</tbody>
</table>

StatusCode Enum
Describes the status code for an unsuccessful component deploy.

**Enum Values**
The following are the values of the Metadata.StatusCode enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>INVALID_SCS_INBOUND_USER</td>
<td>Log in as the RunAs user configured in your SCS setup.</td>
</tr>
</tbody>
</table>
### SCS Connected App

**Value**

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>REQUIRE_CONNECTED_APP_SCS</td>
<td>SCS Connected App is not installed.</td>
</tr>
<tr>
<td>REQUIRE_CONNECTED_APP_SESSION_SCS</td>
<td>To use the SCS connected app, the user must be authenticated.</td>
</tr>
<tr>
<td>REQUIRE_RUNAS_USER</td>
<td>A RunAs user must be configured in your org.</td>
</tr>
</tbody>
</table>

**SEE ALSO:**

- DeployProblemType Enum

### SubtabComponents Class

Represents custom console components on subtabs in the Salesforce console.

**Namespace**

`Metadata`

**Usage**

Use this class when accessing `Metadata.Layout` metadata components. For more information, see “SubtabComponents” in the Metadata API Developer Guide.

**IN THIS SECTION:**

- SubtabComponents Properties
- SubtabComponents Methods

### SubtabComponents Properties

The following are properties for `SubtabComponents`.

**IN THIS SECTION:**

- `component`
  - Represents a custom console component (Visualforce page, lookup field, or related lists) on a section of a page layout.

- `containers`
  - Represents a location and style in which to display more than one custom console component on the sidebars of the Salesforce console.

**component**

Represents a custom console component (Visualforce page, lookup field, or related lists) on a section of a page layout.

**Signature**

```java
public List<Metadata.ConsoleComponent> component {get; set;}
```
Property Value
Type: List<Metadata.ConsoleComponent>

containers
Represents a location and style in which to display more than one custom console component on the sidebars of the Salesforce console.

Signature
public List<Metadata.Container> containers {get; set;}

Property Value
Type: List<Metadata.Container>

SubtabComponents Methods
The following are methods for SubtabComponents.

IN THIS SECTION:
    clone()
    Makes a duplicate copy of the Metadata.SubtabComponents.

    clone()
    Makes a duplicate copy of the Metadata.SubtabComponents.

Signature
public Object clone()

Return Value
Type: Object

SummaryLayoutStyleEnum Enum
Describes the highlights panel style for a SummaryLayout.

Enum Values
The following are the values of the Metadata.SummaryLayoutStyleEnum enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CaseInteraction</td>
<td>Case interaction style.</td>
</tr>
<tr>
<td>ChildServiceReportTemplateStyle</td>
<td>Child service report template style.</td>
</tr>
</tbody>
</table>
### SummaryLayout Class

Controls the appearance of the highlights panel, which summarizes key fields in a grid at the top of a page layout, when Case Feed is enabled.

#### Namespace

*Metadata*

#### Usage

Use this class when accessing `Metadata.Layout` metadata components. For more information, see "SummaryLayout" in the *Metadata API Developer Guide*.

#### SummaryLayout Properties

The following are properties for `SummaryLayout`.

#### SummaryLayout Methods
sizeZ
If provided, the setting is not visible to users.

summaryLayoutItems
Controls the appearance of an individual field and its column and row position within the highlights panel grid, when Case Feed is enabled. At least one is required.

summaryLayoutStyle
Specifies the panel style.

masterLabel
The name of the layout label.

⚠️ Important: Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Signature

class SummaryLayout
{
    public String masterLabel {get; set;}
}

Property Value
Type: String

sizeX
Number of columns in the highlights pane, between 1 and 4 (inclusive).

Signature

class SummaryLayout
{
    public Integer sizeX {get; set;}
}

Property Value
Type: Integer

sizeY
Number of rows in each column, either 1 or 2.

Signature

class SummaryLayout
{
    public Integer sizeY {get; set;}
}

Property Value
Type: Integer

sizeZ
If provided, the setting is not visible to users.
Signature

```java
public Integer sizeZ {get; set;}
```

Property Value

**Type:** Integer

**summaryLayoutItems**

Controls the appearance of an individual field and its column and row position within the highlights panel grid, when Case Feed is enabled. At least one is required.

Signature

```java
public List<Metadata.SummaryLayoutItem> summaryLayoutItems {get; set;}
```

Property Value

**Type:** List<Metadata.SummaryLayoutItem>

**summaryLayoutStyle**

Specifies the panel style.

Signature

```java
public Metadata.SummaryLayoutStyleEnum summaryLayoutStyle {get; set;}
```

Property Value

**Type:** Metadata.SummaryLayoutStyleEnum

**SummaryLayout Methods**

The following are methods for SummaryLayout.

**IN THIS SECTION:**

- **clone()**
  Makes a duplicate copy of the `Metadata.SummaryLayout`.

- **clone()**
  Makes a duplicate copy of the `Metadata.SummaryLayout`.

**Signature**

```java
public Object clone()
```
SummaryLayoutItem Class

Controls the appearance of an individual field and its column and row position within the highlights panel grid, when Case Feed is enabled. You can have two fields per each grid in a highlights panel.

Namespace

Metadata

Usage

Use this class when accessing Metadata.Layout metadata components. For more information, see “SummaryLayoutItem” in the Metadata API Developer Guide.

IN THIS SECTION:

SummaryLayoutItem Properties
SummaryLayoutItem Methods

SummaryLayoutItem Properties

The following are properties for SummaryLayoutItem.

IN THIS SECTION:

customLink
The custom link reference.
field
The field name reference, relative to the page layout. Must be a standard or custom field that also exists on the detail page.
posX
The item’s column position in the highlights panel grid. Must be within the range of sizeX.
posY
The item’s row position in the highlights panel grid. Must be within the range of sizeY.
posZ
Reserved for future use. If provided, the setting is not visible to users.

customLink
The custom link reference.

Signature

public String customLink {get; set;}

2458
Property Value
Type: String

field
The field name reference, relative to the page layout. Must be a standard or custom field that also exists on the detail page.

Signature
public String field {get; set;}

Property Value
Type: String

posX
The item's column position in the highlights panel grid. Must be within the range of sizeX.

Signature
public Integer posX {get; set;}

Property Value
Type: Integer

posY
The item's row position in the highlights panel grid. Must be within the range of sizeY.

Signature
public Integer posY {get; set;}

Property Value
Type: Integer

posZ
Reserved for future use. If provided, the setting is not visible to users.

Signature
public Integer posZ {get; set;}

Property Value
Type: Integer
SummaryLayoutItem Methods

The following are methods for SummaryLayoutItem.

IN THIS SECTION:

clone()

Makes a duplicate copy of the Metadata.SummaryLayoutItem.

clone()

Makes a duplicate copy of the Metadata.SummaryLayoutItem.

Signature

public Object clone()

Return Value

Type: Object

UiBehavior Enum

Describes the behavior for a layout item on a layout page.

Enum Values

The following are the values of the Metadata.UiBehavior enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit</td>
<td>The layout field can be edited but is not required.</td>
</tr>
<tr>
<td>Readonly</td>
<td>The layout field is read-only.</td>
</tr>
<tr>
<td>Required</td>
<td>The layout field can be edited and is required.</td>
</tr>
</tbody>
</table>

Pref_center Namespace

The Pref_center namespace provides an interface, classes, and methods to create and retrieve data in forms in Preference Manager. Preference Manager, previously called Preference Center, is a feature within the Privacy Center app.

The following are the classes in the Pref_center namespace.

IN THIS SECTION:

LoadFormData Class

Retrieve records related to the tokenized record id, and populate the values of a preference form.
LoadParameters Class
Contains methods to retrieve record Id information for parameters passed into the load-form handler.

PreferenceCenterApexHandler Interface
Pass data between your organization and a form in Preference Manager.

SubmitFormData Class
Contains methods to retrieve information on buttons and options selected in a preference form.

SubmitParameters Class
Retrieve record ID information to use with your submit-form handler.

TokenType Enum
Defines the types of values supported by the TokenUtility methods.

TokenUtility Class
Generate authentication tokens to access preference forms.

ValidationResult Class
This class is reserved for future use with Preference Manager.

LoadFormData Class
Retrieve records related to the tokenized record id, and populate the values of a preference form.

Namespace
Pref_center

Example
Use methods in the LoadFormData class to set available and selected values in different form components:

```java
List<System.SelectOption> picklistOptions = new List<System.SelectOption>();
picklistOptions.add(new System.SelectOption('optIn', 'Opt In'));
picklistOptions.add(new System.SelectOption('optOut', 'Opt Out'));

// Set the available options for the picklist
loadFormData.setOptions('myPicklist', picklistOptions);
// Add an option to the existing options for the picklist
loadFormDataдобавимOption('myPicklist', 'optOutAll', 'Opt Out All');
// Select the 'optIn' option in the picklist
loadFormData.setSelectedOption('myPicklist', 'optIn');
```

```java
List<System.SelectOption> checkboxOptions = new List<System.SelectOption>();
checkboxOptions.add(new System.SelectOption('yes', 'Yes'));
checkboxOptions.add(new System.SelectOption('no', 'No'));

// Set available options for the checkbox group
loadFormData.setOptions('myCheckbox', checkboxOptions);
// Select the 'yes' option in the checkbox group
loadFormData.addSelectedOption('myCheckbox', 'yes');
```
// Also select the 'no' option in the checkbox group
loadFormData.addSelectedOption('myCheckbox', 'no');
// Another way to select both the 'yes' and 'no' options in the checkbox group
loadFormData.setSelectedOptions('myCheckbox', new List<String>{'yes', 'no'});

// Fill the value in the text input
loadFormData.setTextValue('myTextInput', 'admin@salesforce.com');
// Set the hint text for the text input
loadFormData.setTextHint('myTextInput', 'Email Address');

// Set the label for the button
loadFormData.setButtonLabel('myButton', 'Save Preferences');

IN THIS SECTION:

LoadFormData Constructors
LoadFormData Methods

LoadFormData Constructors

The following are constructors for LoadFormData.

IN THIS SECTION:

LoadFormData(data)

Creates an instance of the LoadFormData class for running tests on any custom Apex classes you create for Preference Manager.

**LoadFormData (data)**

Creates an instance of the LoadFormData class for running tests on any custom Apex classes you create for Preference Manager.

**Signature**

```java
public LoadFormData(Map<String, pref_center.FieldProperties> data)
```

**Parameters**

`data`

Type: `Map<String, pref_center.FieldProperties>\nMaps preference form data from the field ID to the field properties.

**Usage**

This constructor is available in API version 56.0 and later.

LoadFormData Methods

The following are methods for LoadFormData.
IN THIS SECTION:

- **addOption(fieldId, value, label)**
  Add an option for a checkbox, picklist, or radio button field in a preference form using the label and value.

- **addOption(fieldId, option)**
  Add a defined, selectable option for a checkbox, picklist, or radio button field in a preference form.

- **addSelectedOption(fieldId, option)**
  Add a selected option for a checkbox field in a preference form. This requires the field on the form to have a defined option with a set value.

- **setButtonLabel(fieldId, label)**
  Set the label of a button added to the preference form.

- **setOptions(fieldId, options)**
  Add a list of selectable options for a field on a preference form.

- **setSelectedOption(fieldId, optionValue)**
  For a picklist or radio button field on a preference form that has defined values, set the value entered in the optionValue field as the selected option.

- **setSelectedOptions(fieldId, options)**
  For an existing checkbox field on a preference form that has defined values, set the values entered in the options field as the selected options. This requires the field on the form to have defined options with a set value.

- **setTextHint(fieldId, hintText)**
  Set the hint text inside a text input field. The hint text tells the user what type of information to enter, like an email address.

- **setTextValue(fieldId, value)**
  Set the value of a text field in a preference form.

---

**addOption(fieldId, value, label)**

Add an option for a checkbox, picklist, or radio button field in a preference form using the label and value.

**Signature**

```java
public void addOption(String fieldId, String value, String label)
```

**Parameters**

- **fieldId**
  - Type: `String`
  - Identifies a field in the preference form.

- **value**
  - Type: `String`
  - Represents the selection or text entered in a preference form field.

- **label**
  - Type: `String`
  - The label for the value of a field in a preference form.
Return Value
Type: void

**addOption(fieldId, option)**
Add a defined, selectable option for a checkbox, picklist, or radio button field in a preference form.

**Signature**
```java
public void addOption(String fieldId, System.SelectOption option)
```

**Parameters**
- **fieldId**
  Type: *String*
  Identifies a field in the preference form.
- **option**
  Type: *System.SelectOption*
  The option selected on a field in the preference form.

Return Value
Type: void

**addSelectedOption(fieldId, option)**
Add a selected option for a checkbox field in a preference form. This requires the field on the form to have a defined option with a set value.

**Signature**
```java
public void addSelectedOption(String fieldId, String option)
```

**Parameters**
- **fieldId**
  Type: *String*
  Identifies a field in the preference form.
- **option**
  Type: *String*
  The selectable option being added.

Return Value
Type: void
**setButtonLabel(fieldId, label)**
Set the label of a button added to the preference form.

**Signature**
```
public void setButtonLabel(String fieldId, String label)
```

**Parameters**
- **fieldId**
  - Type: String
  - Identifies a field in the preference form.
- **label**
  - Type: String
  - The label for a button added to the preference form.

**Return Value**
Type: void

**setOptions(fieldId, options)**
Add a list of selectable options for a field on a preference form.

**Signature**
```
public void setOptions(String fieldId, List<SelectOption> options)
```

**Parameters**
- **fieldId**
  - Type: String
  - Identifies a field in the preference form.
- **options**
  - Type: List<SelectOption>
  - The selectable options for a field in the preference form.

**Return Value**
Type: void

**setSelectedOption(fieldId, optionValue)**
For a picklist or radio button field on a preference form that has defined values, set the value entered in the optionValue field as the selected option.
Signature

public void setSelectedOption(String fieldId, String optionValue)

Parameters

fieldId
Type: String
Identifies a field in the preference form.

optionValue
Type: String
The value for the selected option.

Return Value
Type: void

setSelectedOptions(fieldId, options)

For an existing checkbox field on a preference form that has defined values, set the values entered in the options field as the selected options. This requires the field on the form to have defined options with a set value.

Signature

public void setSelectedOptions(String fieldId, List<String> options)

Parameters

fieldId
Type: String
Identifies the checkbox field in the preference form.

options
Type: List<String>
The selected options for a field in the preference form.

Return Value
Type: void

setTextHint(fieldId, hintText)

Set the hint text inside a text input field. The hint text tells the user what type of information to enter, like an email address.

Signature

public void setTextHint(String fieldId, String hintText)
Parameters

**fieldId**
- Type: **String**
  - The ID of the text input field in the preference form.

**hintText**
- Type: **String**
  - The hint text in the text input field.

Return Value

Type: void

**setTextValue(fieldId, value)**
Set the value of a text field in a preference form.

Signature

```java
public void setTextValue(String fieldId, String value)
```

Parameters

**fieldId**
- Type: **String**
  - Identifies a field in the preference form.

**value**
- Type: **String**
  - Represents the value entered for the text field in a preference form.

Return Value

Type: void

LoadParameters Class

Contains methods to retrieve record Id information for parameters passed into the load-form handler.

Namespace

**Pref_center**

Example

```java
String userId = loadParams.getRecordId();

User user = [select id, AboutMe from User where id=:userId];
```
LoadParameters Methods
The following are methods for LoadParameters.

getRecordId()
Returns the untokenized version of the record Id.

Signature
public String getRecordId()

Return Value
Type: String

PreferenceCenterApexHandler Interface
Pass data between your organization and a form in Preference Manager.

Namespace
Pref_center

PreferenceCenterApexHandler Methods
The following are methods for PreferenceCenterApexHandler.

load(loadParams, formData, validationResult)
Retrieve the record IDs and initial values from a preference form before it is edited and submitted.

submit(loadParams, formData, validationResult)
Updates the changed values of fields when the preference form is submitted.
load(loadParams, formData, validationResult)
Retrieves the record IDs and initial values from a preference form before it is edited and submitted.

Signature
public pref_center.LoadFormData load(pref_center.LoadParameters loadParams,
pref_center.LoadFormData formData, pref_center.ValidationResult validationResult)

Parameters
loadParams
  Type: pref_center.LoadParameters
  Retrieve the tokenized record ID.
formData
  Type: pref_center.LoadFormData
  Set the initial values of fields in a form before they are edited.
validationResult
  Type: pref_center.ValidationResult
  Reserved for future use.

Return Value
Type: pref_center.LoadFormData

submit(loadParams, formData, validationResult)
Updates the changed values of fields when the preference form is submitted.

Signature
public void submit(pref_center.SubmitParameters submitParams,
pref_center.SubmitFormData formData, pref_center.ValidationResult validationResult)

Parameters
submitParams
  Type: pref_center.SubmitParameters
  Retrieve the tokenized record ID.
formData
  Type: pref_center.SubmitFormData
  Retrieve the values of fields in a submitted form.
validationResult
  Type: pref_center.ValidationResult
  Reserved for future use.
Return Value
Type: void

SubmitFormData Class
Contains methods to retrieve information on buttons and options selected in a preference form.

Namespace
Pref_center

Example
Use methods in the SubmitFormData class to retrieve the selected values in different form components:

```java
String buttonClickedId = formData.getButtonClicked();
if (buttonClickedId == 'submitButton') {
    // Handle form submit
} else if (buttonClickedId == 'cancelButton') {
    // Handle form cancel
}

String picklistValueOld = formData.getOldSelectedValue('myPicklist');
String picklistValueNew = formData.getSelectedValue('myPicklist');
if (picklistValueOld != picklistValueNew) {
    // Do something
}

List<String> checkboxValuesOld = formData.getOldSelectedValues('myCheckbox');
List<String> checkboxValuesNew = formData.getSelectedValues('myCheckbox');
if (checkboxValuesOld != null && checkboxValuesNew != null && (checkboxValuesOld.size() != checkboxValuesNew.size())) {
    // Do something
}

String textinputValueOld = formData.getOldStringValue('myTextinput');
String textinputValueNew = formData.getStringValue('myTextinput');
if (textinputValueOld != textinputValueNew) {
    // Do something
}
```

IN THIS SECTION:
SubmitFormData Methods

SubmitFormData Methods
The following are methods for SubmitFormData.
IN THIS SECTION:

getButtonClicked()
Returns the field ID of the button that was clicked in the preference form. For example, use this method to determine if the clicked button was Submit or Cancel.

getOldSelectedValue(fieldId)
Returns the value that was set for the specified field when the preference form was previously edited by the user. This method is used for field types such as picklist or radio buttons.

getOldSelectedValues(fieldId)
Returns a list of the string values that were set on a checkbox field when the preference form was previously edited by the user.

getOldStringValue(fieldId)
Returns the string value that was set on a field when the preference form was loaded. This method is used for field types such as text, and throws a TypeException if used with a field that can return more than one value, like a checkbox field.

getSelectedValue(fieldId)
Returns the string value that is currently selected for a picklist or radio button field in the preference form.

getSelectedValues(fieldId)
Returns a list of string values that are currently selected on a checkbox field in the preference form.

getStringValue(fieldId)
Returns the string value that was set on a field when the preference form was loaded. This method is used for field types such as text.

---

**getButtonClicked()**

Returns the field ID of the button that was clicked in the preference form. For example, use this method to determine if the clicked button was Submit or Cancel.

**Signature**

```java
public String getButtonClicked()
```

**Return Value**

Type: String

**getOldSelectedValue(fieldId)**

Returns the value that was set for the specified field when the preference form was previously edited by the user. This method is used for field types such as picklist or radio buttons.

**Signature**

```java
public String getOldSelectedValue(String fieldId)
```

**Parameters**

- **fieldId**
  Type: String

---

2471
Identifies a field in the preference form.

Return Value
Type: String

`getOldSelectedValues(fieldId)`
Returns a list of the string values that were set on a checkbox field when the preference form was previously edited by the user.

Signature
```java
public List<String> getOldSelectedValues(String fieldId)
```

Parameters
`fieldId`
Type: String
Identifies a field in the preference form.

Return Value
Type: List<String>

`getOldStringValue(fieldId)`
Returns the string value that was set on a field when the preference form was loaded. This method is used for field types such as text, and throws a TypeException if used with a field that can return more than one value, like a checkbox field.

Signature
```java
public String getOldStringValue(String fieldId)
```

Parameters
`fieldId`
Type: String
Identifies a field in the preference form.

Return Value
Type: String

`getSelectedValue(fieldId)`
Returns the string value that is currently selected for a picklist or radio button field in the preference form.

Signature
```java
public String getSelectedValue(String fieldId)
```
Parameters

fieldId
Type: String
Identifies a field in the preference form.

Return Value
Type: String

getSelectedValues(fieldId)
Returns a list of string values that are currently selected on a checkbox field in the preference form.

Signature
public List<String> getSelectedValues(String fieldId)

Parameters

fieldId
Type: String
Identifies a field in the preference form.

Return Value
Type: List<String>

getStringValue(fieldId)
Returns the string value that was set on a field when the preference form was loaded. This method is used for field types such as text.

Signature
public String getStringValue(String fieldId)

Parameters

fieldId
Type: String
Identifies a field in the preference form.

Return Value
Type: String

SubmitParameters Class
Retrieve record ID information to use with your submit-form handler.
### Namespace

Pref_center

### Example

```java
String userId = submitParams.getRecordId();
User user = [select id, AboutMe from User where id=:userId];
```

IN THIS SECTION:

SubmitParameters Methods

### SubmitParameters Methods

The following are methods for SubmitParameters.

IN THIS SECTION:

getRecordId()

Returns the untokenized version of the record ID.

---

#### getRecordId()

Returns the untokenized version of the record ID.

---

### Signature

`public String getRecordId()`

---

### Return Value

Type: `String`

---

### TokenType Enum

Defines the types of values supported by the TokenUtility methods.

---

### Enum Values

The following are the values of the `pref_center.TokenType` enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMAIL</td>
<td>Identifies the token as an email address.</td>
</tr>
<tr>
<td>STANDARD</td>
<td>Identifies the token as a Salesforce record ID. This is the default token type.</td>
</tr>
</tbody>
</table>
TokenUtility Class

Generate authentication tokens to access preference forms.

Namespace

Pref_center

Example

Call the `generateToken()` method to generate a single token for a specified Salesforce record ID:

``` apex
Individual individual = [SELECT Id FROM Individual LIMIT 1];
String token = pref_center.TokenUtility.generateToken(individual.Id);
// Do something with the token
System.debug(token)
```

Call the `generateTokens()` method to generate tokens in bulk when given a list of Salesforce record IDs:

``` apex
List<Id> individualIds = new List<Id>();
// Get Ids of Individuals who have not opted out of tracking
for (Individual individual : [SELECT Id FROM Individual WHERE HasOptedOutTracking = false])
    individualIds.add(individual.Id);
// Generate tokens for the list of Individual record Ids
Map<String, String> tokens = pref_center.TokenUtility.generateTokens(individualIds);
String firstIndividualId = individualIds[0];
// The returned Map has the input record Id as key and the corresponding token as value
String tokenForFirstIndividual = tokens.get(firstIndividualId);
// Do something with the token
System.debug(tokenForFirstIndividual);
```

IN THIS SECTION:

TokenUtility Methods

TokenUtility Methods

The following are methods for `TokenUtility`.

IN THIS SECTION:

- `generateToken(tokenValue, tokenType)`
  Returns the authentication token for the specified token value using the given token type.
- `generateToken(tokenValue)`
  Returns the authentication token for the specified token value using the default `standard` token type.
- `generateTokens(tokenValues, tokenType)`
  Returns the authentication tokens in the form of a map, where the map key is the input value to be tokenized and the map value is the corresponding token. The given token type is used to generate the tokens.
generateTokens(tokenValues)
Returns the generated tokens in the form of a map. This method uses the default standard token type to generate the tokens.

generateToken(tokenValue, tokenType)
Returns the authentication token for the specified token value using the given token type.

Signature
public static String generateToken(String tokenValue, pref_center.TokenType tokenType)

Parameters
tokenValue
Type: String
The value passed to LoadParameters.getRecordId() and SubmitParameters.getRecordId(). Identifies the entity that the preference form is acting on.
tokenType
Type: pref_center.TokenType
Specifies the type of the value to be encrypted with authentication tokens.

Return Value
Type: String

generateToken(tokenValue)
Returns the authentication token for the specified token value using the default standard token type.

Signature
public static String generateToken(String tokenValue)

Parameters
tokenValue
Type: String
Identifies the entity that the preference form is acting on. The value passed to LoadParameters.getRecordId() and SubmitParameters.getRecordId().

Return Value
Type: String

generateTokens(tokenValues, tokenType)
Returns the authentication tokens in the form of a map, where the map key is the input value to be tokenized and the map value is the corresponding token. The given token type is used to generate the tokens.
**Signature**

```java
public static Map<String,String> generateTokens(List<String> tokenValues,
pref_center.TokenType tokenType)
```

**Parameters**

- **tokenValues**
  - Type: `List<String>`
  - The values passed to `LoadParameters.getRecordId()` and `SubmitParameters.getRecordId()`. Identifies the entity that the preference form is acting on. Contains multiple values to be encrypted with authentication tokens.

- **tokenType**
  - Type: `pref_center.TokenType`
  - Specifies the type of the value to be encrypted with authentication tokens.

**Return Value**

- Type: `Map<String,String>`, where the map key is the input value to be tokenized and the map value is the corresponding token.

**ValidationResult Class**

This class is reserved for future use with Preference Manager.

**Namespace**

`Pref_center`
Process Namespace

The Process namespace provides an interface and classes for passing data between your organization and a flow. The following are the interfaces and classes in the Process namespace.

IN THIS SECTION:

- Plugin Interface
- PluginDescribeResult Class
- PluginDescribeResult.InputParameter Class
- PluginDescribeResult.OutputParameter Class
- PluginRequest Class
- PluginResult Class

Plugin Interface

Allows you to pass data between your organization and a specified flow.

Namespace

Process

Tip: We recommend using the @InvocableMethod annotation instead of the Process.Plugin interface.

- The interface doesn’t support Blob, Collection, sObject, and Time data types, and it doesn’t support bulk operations. Once you implement the interface on a class, the class can be referenced only from flows.
- The annotation supports all data types and bulk operations. Once you implement the annotation on a class, the class can be referenced from flows, processes, and the Custom Invocable Actions REST API endpoint.

IN THIS SECTION:

- Plugin Methods
- Plugin Example Implementation

Plugin Methods

The following are instance methods for Plugin.
IN THIS SECTION:

`describe()`

Returns a `Process.PluginDescribeResult` object that describes this method call.

`invoke(request)`

Primary method that the system invokes when the class that implements the interface is instantiated.

**describe()**

Returns a `Process.PluginDescribeResult` object that describes this method call.

**Signature**

```
public Process.PluginDescribeResult describe()
```

**Return Value**

Type: `Process.PluginDescribeResult`

**invoke(request)**

Primary method that the system invokes when the class that implements the interface is instantiated.

**Signature**

```
```

**Parameters**

`request`

Type: `Process.PluginRequest`

**Return Value**

Type: `Process.PluginResult`

**Plugin Example Implementation**

```java
global class flowChat implements Process.Plugin {

    // The main method to be implemented. The Flow calls this at run time.
    global Process.PluginResult invoke(Process.PluginRequest request) {
        // Get the subject of the Chatter post from the flow
        String subject = (String) request.inputParameters.get('subject');

        // Use the Chatter APIs to post it to the current user's feed
        FeedItem fItem = new FeedItem();
        fItem.ParentId = UserInfo.getUserId();
        fItem.Body = 'Flow Update: ' + subject;
        insert fItem;
    }
}
```
// return to Flow
Map<String, Object> result = new Map<String, Object>();
return new Process.PluginResult(result);
}

// Returns the describe information for the interface
global Process.PluginDescribeResult describe() {
    Process.PluginDescribeResult result = new Process.PluginDescribeResult();
    result.Name = 'flowchatplugin';
    result.Tag = 'chat';
    result.inputParameters = new List<Process.PluginDescribeResult.InputParameter>{
        new Process.PluginDescribeResult.InputParameter('subject',
            Process.PluginDescribeResult.ParameterType.STRING, true)
    };
    result.outputParameters = new List<Process.PluginDescribeResult.OutputParameter>{}
    return result;
}

Test Class
The following is a test class for the above class.

@isTest
private class flowChatTest {

    static testmethod void flowChatTests() {
        flowChat plugin = new flowChat();
        Map<String, Object> inputParams = new Map<String, Object>();

        string feedSubject = 'Flow is alive';
        InputParams.put('subject', feedSubject);

        Process.PluginRequest request = new Process.PluginRequest(inputParams);

        plugin.invoke(request);
    }
}

PluginDescribeResult Class
Describes the input and output parameters for Process.PluginResult.

Namespace
Process
Tip: We recommend using the @InvocableMethod annotation instead of the Process.Plugin interface.

- The interface doesn’t support Blob, Collection, sObject, and Time data types, and it doesn’t support bulk operations. Once you implement the interface on a class, the class can be referenced only from flows.
- The annotation supports all data types and bulk operations. Once you implement the annotation on a class, the class can be referenced from flows, processes, and the Custom Invocable Actions REST API endpoint.

IN THIS SECTION:
- PluginDescribeResult Constructors
- PluginDescribeResult Properties

PluginDescribeResult Constructors

The following are constructors for PluginDescribeResult.

IN THIS SECTION:
- PluginDescribeResult()

PluginDescribeResult()

Creates a new instance of the Process.PluginDescribeResult class.

Signature

public PluginDescribeResult()

PluginDescribeResult Properties

The following are properties for PluginDescribeResult.

IN THIS SECTION:
- description
- inputParameters
- name
- outputParameters

description

This optional field describes the purpose of the plug-in.

inputParameters

The input parameters passed by the Process.PluginRequest class from a flow to the class that implements the Process.Plugin interface.

name

Unique name of the plug-in.

outputParameters

The output parameters passed by the Process.PluginResult class from the class that implements the Process.Plugin interface to the flow.
**description**
This optional field describes the purpose of the plug-in.

**Signature**

```java
public String description {get; set;}
```

**Property Value**
Type: String

**Usage**
Size limit: 255 characters.

**inputParameters**
The input parameters passed by the `Process.PluginRequest` class from a flow to the class that implements the `Process.Plugin` interface.

**Signature**

```java
public List<Process.PluginDescribeResult.InputParameter> inputParameters {get; set;}
```

**Property Value**
Type: `List<Process.PluginDescribeResult.InputParameter>`

**name**
Unique name of the plug-in.

**Signature**

```java
public String name {get; set;}
```

**Property Value**
Type: String

**Usage**
Size limit: 40 characters.

**outputParameters**
The output parameters passed by the `Process.PluginResult` class from the class that implements the `Process.Plugin` interface to the flow.
Signature
public List<Process.PluginDescribeResult.OutputParameter> outputParameters {get; set;}

Property Value
Type: List<Process.PluginDescribeResult.OutputParameter>

PluginDescribeResult.InputParameter Class
Describes the input parameter for Process.PluginResult.

Namespace
Process

Tip: We recommend using the @InvocableMethod annotation instead of the Process.Plugin interface.
- The interface doesn’t support Blob, Collection, sObject, and Time data types, and it doesn’t support bulk operations. Once you implement the interface on a class, the class can be referenced only from flows.
- The annotation supports all data types and bulk operations. Once you implement the annotation on a class, the class can be referenced from flows, processes, and the Custom Invocable Actions REST API endpoint.

IN THIS SECTION:
- PluginDescribeResult.InputParameter Constructors
- PluginDescribeResult.InputParameter Properties

PluginDescribeResult.InputParameter Constructors
The following are constructors for PluginDescribeResult.InputParameter.

IN THIS SECTION:
- PluginDescribeResult.InputParameter(name, description, parameterType, required)
  Creates a new instance of the Process.PluginDescribeResult.InputParameter class using the specified name, description, parameter type, and required option.
- PluginDescribeResult.InputParameter(name, parameterType, required)
  Creates a new instance of the Process.PluginDescribeResult.InputParameter class using the specified name, parameter type, and required option.

PluginDescribeResult.InputParameter(name, description, parameterType, required)
Creates a new instance of the Process.PluginDescribeResult.InputParameter class using the specified name, description, parameter type, and required option.
**Signature**

```java
public PluginDescribeResult.InputParameter(String name, String description,
Process.PluginDescribeResult.ParameterType parameterType, Boolean required)
```

**Parameters**

- **name**
  - Type: `String`
  - Unique name of the plug-in.

- **description**
  - Type: `String`
  - Describes the purpose of the plug-in.

- **parameterType**
  - Type: `Process.PluginDescribeResult.ParameterType`
  - The data type of the input parameter.

- **required**
  - Type: `Boolean`
  - Set to `true` for required and `false` otherwise.

**PluginDescribeResult.InputParameter(name, parameterType, required)**

Creates a new instance of the `Process.PluginDescribeResult.InputParameter` class using the specified name, parameter type, and required option.

**PluginDescribeResult.InputParameter Properties**

The following are properties for `PluginDescribeResult.InputParameter`. No specific properties are listed in the provided text.
IN THIS SECTION:

- **Description**
  This optional field describes the purpose of the plug-in.

- **Name**
  Unique name of the plug-in.

- **ParameterType**
  The data type of the input parameter.

- **Required**
  Set to `true` for required and `false` otherwise.

**Description**
This optional field describes the purpose of the plug-in.

**Signature**

```csharp
public String Description {get; set;}
```

**Property Value**
Type: `String`

**Usage**
Size limit: 255 characters.

**Name**
Unique name of the plug-in.

**Signature**

```csharp
public String Name {get; set;}
```

**Property Value**
Type: `String`

**Usage**
Size limit: 40 characters.

**ParameterType**
The data type of the input parameter.

**Signature**

```csharp
public Process.PluginDescribeResult.ParameterType ParameterType {get; set;}
```
Property Value
Type: Process.PluginDescribeResult.ParameterType

Required
Set to true for required and false otherwise.

Signature
public Boolean Required {get; set;}

Property Value
Type: Boolean

PluginDescribeResult.OutputParameter Class
Describes the output parameter for Process.PluginResult.

Namespace
Process

Tip: We recommend using the @InvocableMethod annotation instead of the Process.Plugin interface.

- The interface doesn’t support Blob, Collection, sObject, and Time data types, and it doesn’t support bulk operations. Once you implement the interface on a class, the class can be referenced only from flows.
- The annotation supports all data types and bulk operations. Once you implement the annotation on a class, the class can be referenced from flows, processes, and the Custom Invocable Actions REST API endpoint.

IN THIS SECTION:
- PluginDescribeResult.OutputParameter Constructors
- PluginDescribeResult.OutputParameter Properties

PluginDescribeResult.OutputParameter Constructors
The following are constructors for PluginDescribeResult.OutputParameter.

IN THIS SECTION:
- PluginDescribeResult.OutputParameter(name, description, parameterType)
  Creates a new instance of the Process.PluginDescribeResult.OutputParameter class using the specified name, description, and parameter type.
- PluginDescribeResult.OutputParameter(name, parameterType)
  Creates a new instance of the Process.PluginDescribeResult.OutputParameter class using the specified name, description, and parameter type.
PluginDescribeResult.OutputParameter(name, description, parameterType)

Creates a new instance of the Process.PluginDescribeResult.OutputParameter class using the specified name, description, and parameter type.

**Signature**

```java
public PluginDescribeResult.OutputParameter(String name, String description,
Process.PluginDescribeResult.ParameterType parameterType)
```

**Parameters**

- **name**
  - Type: String
  - Unique name of the plug-in.

- **description**
  - Type: String
  - Describes the purpose of the plug-in.

- **parameterType**
  - Type: Process.PluginDescribeResult.ParameterType
  - The data type of the input parameter.

PluginDescribeResult.OutputParameter(name, parameterType)

Creates a new instance of the Process.PluginDescribeResult.OutputParameter class using the specified name, description, and parameter type.

**Signature**

```java
public PluginDescribeResult.OutputParameter(String name,
Process.PluginDescribeResult.ParameterType parameterType)
```

**Parameters**

- **name**
  - Type: String
  - Unique name of the plug-in.

- **parameterType**
  - Type: Process.PluginDescribeResult.ParameterType
  - The data type of the input parameter.

**PluginDescribeResult.OutputParameter Properties**

The following are properties for PluginDescribeResult.OutputParameter.
IN THIS SECTION:

**Description**
This optional field describes the purpose of the plug-in.

**Name**
Unique name of the plug-in.

**ParameterType**
The data type of the input parameter.

**Description**
This optional field describes the purpose of the plug-in.

**Signature**

```csharp
public String Description {get; set;}
```

**Property Value**
Type: String

**Usage**
Size limit: 255 characters.

**Name**
Unique name of the plug-in.

**Signature**

```csharp
public String Name {get; set;}
```

**Property Value**
Type: String

**Usage**
Size limit: 40 characters.

**ParameterType**
The data type of the input parameter.

**Signature**

```csharp
public Process.PluginDescribeResult.ParameterType ParameterType {get; set;}
```
Property Value
Type: Process.PluginDescribeResult.ParameterType

PluginRequest Class
Passes input parameters from the class that implements the Process.Plugin interface to the flow.

Namespace
Process

Tip: We recommend using the @InvocableMethod annotation instead of the Process.Plugin interface.
  - The interface doesn’t support Blob, Collection, sObject, and Time data types, and it doesn’t support bulk operations. Once you implement the interface on a class, the class can be referenced only from flows.
  - The annotation supports all data types and bulk operations. Once you implement the annotation on a class, the class can be referenced from flows, processes, and the Custom Invocable Actions REST API endpoint.

PluginRequest Properties
The following are properties for PluginRequest.

IN THIS SECTION:
inputParameters
Input parameters that are passed from the class that implements the Process.Plugin interface to the flow.

inputParameters
Input parameters that are passed from the class that implements the Process.Plugin interface to the flow.

Signature
public MAP<String,ANY> inputParameters {get; set;}

Property Value
Type: Map<String, Object>

PluginResult Class
Returns output parameters from the class that implements the Process.Plugin interface to the flow.

Tip: We recommend using the @InvocableMethod annotation instead of the Process.Plugin interface.
  - The interface doesn’t support Blob, Collection, sObject, and Time data types, and it doesn’t support bulk operations. Once you implement the interface on a class, the class can be referenced only from flows.
  - The annotation supports all data types and bulk operations. Once you implement the annotation on a class, the class can be referenced from flows, processes, and the Custom Invocable Actions REST API endpoint.
Namespace

Process

PluginResult Properties

The following are properties for PluginResult.

IN THIS SECTION:

outputParameters
Output parameters returned from the class that implements the interface to the flow.

outputParameters
Output parameters returned from the class that implements the interface to the flow.

Signature

public MAP<String, ANY> outputParameters {get; set;}

Property Value

Type: Map<String, Object>

QuickAction Namespace

The QuickAction namespace provides classes and methods for quick actions.

The following are the classes in the QuickAction namespace.

IN THIS SECTION:

DescribeAvailableQuickActionResult Class
Contains describe metadata information for a quick action that is available for a specified parent.

DescribeLayoutComponent Class
Represents the smallest unit in a layout—a field or a separator.

DescribeLayoutItem Class
Represents an individual item in a QuickAction.DescribeLayoutRow.

DescribeLayoutRow Class
Represents a row in a QuickAction.DescribeLayoutSection.

DescribeLayoutSection Class
Represents a section of a layout and consists of one or more columns and one or more rows (an array of QuickAction.DescribeLayoutRow).

DescribeQuickActionDefaultValue Class
Returns a default value for a quick action.
DescribeQuickActionResult Class
Contains describe metadata information for a quick action.

QuickActionDefaults Class
Represents an abstract Apex class that provides the context for running the standard Email Action on Case Feed and the container of the Email Message fields for the action payload. You can override the target fields before the standard Email Action is rendered.

QuickActionDefaultsHandler Interface
The QuickAction.QuickActionDefaultsHandler interface lets you specify the default values for the standard Email and Send Email actions in the case feed. You can use this interface to specify the From address, CC address, BCC address, subject, and email body for the Email action in the case feed. You can use the interface to pre-populate these fields based on the context where the action is displayed, such as the case origin (for example, country) and subject.

QuickActionRequest Class
Use the QuickAction.QuickActionRequest class for providing action information for quick actions to be performed by QuickAction class methods. Action information includes the action name, context record ID, and record.

QuickActionResult Class
After you initiate a quick action with the QuickAction class, use the QuickActionResult class for processing action results.

SendEmailQuickActionDefaults Class
Represents an Apex class that provides: the From address list; the original email’s email message ID, provided that the reply action was invoked on the email message feed item; and methods to specify related settings on templates. You can override these fields before the standard Email Action is rendered.

DescribeAvailableQuickActionResult Class
Contains describe metadata information for a quick action that is available for a specified parent.

Namespace
QuickAction

Usage
The QuickAction describeAvailableQuickActions method returns an array of available quick action describe result objects (QuickAction.DescribeAvailableQuickActionResult).

DescribeAvailableQuickActionResult Methods
The following are methods for DescribeAvailableQuickActionResult. All are instance methods.

IN THIS SECTION:

getActionEnumOrId()
Returns the unique ID for the action. If the action doesn’t have an ID, its API name is used.

getLabel()
The quick action label.
getName()  
The quick action name.

getType()  
The quick action type.

getActionEnumOrId()  
Returns the unique ID for the action. If the action doesn’t have an ID, its API name is used.

Signature

public String getActionEnumOrId()

Return Value

Type: String

getLabel()  
The quick action label.

Signature

public String getLabel()

Return Value

Type: String

getName()  
The quick action name.

Signature

public String getName()

Return Value

Type: String

getType()  
The quick action type.

Signature

public String getType()
Return Value
Type: String

DescribeLayoutComponent Class
Represents the smallest unit in a layout—a field or a separator.

Namespace
QuickAction

DescribeLayoutComponent Methods
The following are methods for DescribeLayoutComponent. All are instance methods.

IN THIS SECTION:
- getDisplayLines()
  Returns the vertical lines displayed for a field. Applies to textarea and multi-select picklist fields.
- getTabOrder()
  Returns the tab order for the item in the row.
- getType()
  Returns the name of the QuickAction.DescribeLayoutComponent type for this component.
- getValue()
  Returns the name of the field if the type for QuickAction.DescribeLayoutComponent is textarea.

getDisplayLines()
Returns the vertical lines displayed for a field. Applies to textarea and multi-select picklist fields.

Signature
public Integer getDisplayLines()

Return Value
Type: Integer

getTabOrder()
Returns the tab order for the item in the row.

Signature
public Integer getTabOrder()
Return Value
Type: Integer

getType()  
Returns the name of the QuickAction.DescribeLayoutComponent type for this component.

Signature
public String getType()

Return Value
Type: String

getValue()  
Returns the name of the field if the type for QuickAction.DescribeLayoutComponent is textarea.

Signature
public String getValue()

Return Value
Type: String

DescribeLayoutItem Class
Represents an individual item in a QuickAction.DescribeLayoutRow.

Namespace
QuickAction

Usage
For most fields on a layout, there is only one component per layout item. However, in a display-only view, the QuickAction.DescribeLayoutItem might be a composite of the individual fields (for example, an address can consist of street, city, state, country, and postal code data). On the corresponding edit view, each component of the address field would be split up into separate QuickAction.DescribeLayoutItems.

DescribeLayoutItem Methods
The following are methods for DescribeLayoutItem. All are instance methods.
IN THIS SECTION:

**getLabel()**
Returns the label text for this item.

**getLayoutComponents()**
Returns a list of `QuickAction.DescribeLayoutComponents` for this item.

**isEditableForNew()**
Indicates whether this item can be edited for new (`true`) or not (`false`).

**isEditableForUpdate()**
Indicates whether this item can be edited for update (`true`) or not (`false`).

**isPlaceholder()**
Indicates whether this item is a placeholder (`true`) or not (`false`). If `true`, then this item is blank.

**isRequired()**
Indicates whether this item is required (`true`) or not (`false`).

---

**getLabel()**
Returns the label text for this item.

**Signature**

```java
public String getLabel()
```

**Return Value**

Type: `String`

---

**getLayoutComponents()**
Returns a list of `QuickAction.DescribeLayoutComponents` for this item.

**Signature**

```java
public List<QuickAction.DescribeLayoutComponent> getLayoutComponents()
```

**Return Value**

Type: `List<QuickAction.DescribeLayoutComponent>`

---

**isEditableForNew()**
Indicates whether this item can be edited for new (`true`) or not (`false`).

**Signature**

```java
public Boolean isEditableForNew()
```
Return Value
Type: Boolean

**isEditableForUpdate()**
Indicates whether this item can be edited for update (true) or not (false).

**Signature**
```
public Boolean isEditableForUpdate()
```

Return Value
Type: Boolean

**isPlaceholder()**
Indicates whether this item is a placeholder (true) or not (false). If true, then this item is blank.

**Signature**
```
public Boolean isPlaceholder()
```

Return Value
Type: Boolean

**isRequired()**
Indicates whether this item is required (true) or not (false).

**Signature**
```
public Boolean isRequired()
```

Return Value
Type: Boolean

**Usage**
This is useful if, for example, you want to render required fields in a contrasting color.

**DescribeLayoutRow Class**
Represents a row in a `QuickAction.DescribeLayoutSection`.

**Namespace**
`QuickAction`
Usage

A QuickAction.DescribeLayoutRow consists of one or more QuickAction.DescribeLayoutItem objects. For each QuickAction.DescribeLayoutRow, a QuickAction.DescribeLayoutItem refers either to a specific field or to an "empty" QuickAction.DescribeLayoutItem (one that contains no QuickAction.DescribeLayoutComponent objects). An empty QuickAction.DescribeLayoutItem can be returned when a given QuickAction.DescribeLayoutRow is sparse (for example, containing more fields on the right column than on the left column).

DescribeLayoutRow Methods

The following are methods for DescribeLayoutRow. All are instance methods.

IN THIS SECTION:

getLayoutItems()
Returns either a specific field or an empty QuickAction.DescribeLayoutItem (one that contains no QuickAction.DescribeLayoutComponent objects).

getNumItems()
Returns the number of QuickAction.DescribeLayoutItem.

getLayoutItems()

Returns either a specific field or an empty QuickAction.DescribeLayoutItem (one that contains no QuickAction.DescribeLayoutComponent objects).

Signature

public List<QuickAction.DescribeLayoutItem> getLayoutItems()

Return Value
Type: List<QuickAction.DescribeLayoutItem>

getNumItems()

Returns the number of QuickAction.DescribeLayoutItem.

Signature

public Integer getNumItems()

Return Value
Type: Integer

DescribeLayoutSection Class

Represents a section of a layout and consists of one or more columns and one or more rows (an array of QuickAction.DescribeLayoutRow).
Namespace

QuickAction

DescribeLayoutSection Properties

The following are properties for DescribeLayoutSection.

collapsed

The current view of the record details section: collapsed (true) or expanded (false).

Signature

public Boolean collapsed {get; set;}

Property Value

Type: Boolean

layoutsectionid

The unique ID of the record details section in the layout.

Signature

public Id layoutsectionid {get; set;}

Property Value

Type: Id

DescribeLayoutSection Methods

The following are methods for DescribeLayoutSection.

IN THIS SECTION:

getColumns()  
Returns the number of columns in the QuickAction.DescribeLayoutSection.

getHeading()  
The heading text (label) for the QuickAction.DescribeLayoutSection.

getLayoutRows()  
Returns an array of one or more QuickAction.DescribeLayoutRow objects.

getLayoutSectionId()  
Returns the ID of the record details section in the layout.

getParentLayoutId()  
Returns the ID of the layout upon which this DescribeLayoutSection resides.
getRows()
Returns the number of rows in the QuickAction.DescribeLayoutSection.

isCollapsed()
Indicates whether the record details section is collapsed (true) or expanded (false). If you build your own app, you can use this method to see whether the current user collapsed a section, and respect that preference in your own UI.

isUseCollapsibleSection()
Indicates whether the QuickAction.DescribeLayoutSection is a collapsible section (true) or not (false).

isUseHeading()
Indicates whether to use the heading (true) or not (false).

getColumns()
Returns the number of columns in the QuickAction.DescribeLayoutSection.

Signature
public Integer getColumns()

Return Value
Type: Integer

getHeading()
The heading text (label) for the QuickAction.DescribeLayoutSection.

Signature
public String getHeading()

Return Value
Type: String

getLayoutRows()
Returns an array of one or more QuickAction.DescribeLayoutRow objects.

Signature
public List<QuickAction.DescribeLayoutRow> getLayoutRows()

Return Value
Type: List<QuickAction.DescribeLayoutRow>

getLayoutSectionId()
Returns the ID of the record details section in the layout.
Signature

```java
public Id getLayoutSectionId()
```

Return Value

Type: Id

```java
public Id getParentLayoutId()
```

Returns the ID of the layout upon which this DescribeLayoutSection resides.

Return Value

Type: Id

```java
public Integer getRows()
```

Returns the number of rows in the QuickAction.DescribeLayoutSection.

Return Value

Type: Integer

```java
public Boolean isCollapsed()
```

Indicates whether the record details section is collapsed (true) or expanded (false). If you build your own app, you can use this method to see whether the current user collapsed a section, and respect that preference in your own UI.

Return Value

Type: Boolean

```java
public Boolean isUseCollapsibleSection()
```

Indicates whether the QuickAction.DescribeLayoutSection is a collapsible section (true) or not (false).
Signature

public Boolean isUseCollapsibleSection()

Return Value

Type: Boolean

isUseHeading()

Indicates whether to use the heading (true) or not (false).

Signature

public Boolean isUseHeading()

Return Value

Type: Boolean

DescribeQuickActionDefaultValue Class

Returns a default value for a quick action.

Namespace

QuickAction

Usage

Represents the default values of fields to use in default layouts.

DescribeQuickActionDefaultValue Methods

The following are methods for DescribeQuickActionDefaultValue. All are instance methods.

IN THIS SECTION:

getDefaultValue()

Returns the default value of the quick action.

getField()

Returns the field name of the action.

give the default value of the quick action.
Signature

public String getDefaultValue()

Return Value
Type: String

gField()

Returns the field name of the action.

Signature

public String getField()

Return Value
Type: String

DescribeQuickActionResult Class

Contains describe metadata information for a quick action.

Namespace

QuickAction

Usage

The QuickAction describeQuickActions method returns an array of quick action describe result objects (QuickAction.DescribeQuickActionResult).

IN THIS SECTION:
- DescribeQuickActionResult Properties
- DescribeQuickActionResult Methods

DescribeQuickActionResult Properties

The following are properties for DescribeQuickActionResult.

IN THIS SECTION:
- canvasapplicationname
  The name of the Canvas application invoked by the custom action.
- colors
  Array of color information. Each color is associated with a theme.
contextobjecttype
The object used for the action. Was getsourceSobjectType() in API version 29.0 and earlier.

defaultvalues
The action’s default values.

flowdevname
If the custom action invokes a flow, the fully qualified name of the flow.

flowrecordidvar
If the custom action invokes a flow, the input variable that the custom action passes the record’s ID to.

height
The height in pixels of the action pane.

iconname
The name of the icon used for the action. If a custom icon is not used, this value isn’t set.

icons
Array of icons. Each icon is associated with a theme.

iconurl
The URL of the icon used for the action. This icon URL corresponds to the 32x32 icon used for the current Salesforce theme, introduced in Spring ’10, or the custom icon, if there is one.

layout
The section of the layout where the action resides.

lightningcomponentbundleid
If the custom action invokes a Lightning component, the ID of the Lightning component bundle to which the component belongs.

lightningcomponentbundlename
If the custom action invokes a Lightning component, the name of the Lightning component bundle to which the component belongs.

lightningcomponentqualifiedname
The fully qualified name of the Lightning component invoked by the custom action.

miniiconurl
The icon’s URL. This icon URL corresponds to the 16x16 icon used for the current Salesforce theme, introduced in Spring ’10, or the custom icon, if there is one.

showquickactionlcheader
Indicates whether the Lightning component quick action header and footer are shown. If false, then both the header containing the quick action title and the footer containing the Save and Cancel buttons aren’t displayed.

showquickactionvfheader
Indicates whether the Visualforce quick action header and footer should be shown. If false, then both the header containing the quick action title and the footer containing the Save and Cancel buttons aren’t displayed.

targetparentfield
The parent object type of the action. Links the target object to the parent object. For example, the value is Account if the target object is Contact and the parent object is Account.

targetrecordtypeid
The record type of the target record.
targetsobjecttype
The action's target object type.

visualforcepagename
The name of the Visualforce page associated with the custom action.

visualforcepageurl
The URL of the Visualforce page associated with the action.

width
The width in pixels of the action pane, for custom actions that call Visualforce pages, Canvas apps, or Lightning components.

canvasapplicationname
The name of the Canvas application invoked by the custom action.

Signature
public String canvasapplicationname {get; set;}

Property Value
Type: String

colors
Array of color information. Each color is associated with a theme.

Signature
public List<Schema.DescribeColorResult> colors {get; set;}

Property Value
Type: List<Schema.DescribeColorResult> on page 2707

contextsobjecttype
The object used for the action. Was getsourceSobjectType() in API version 29.0 and earlier.

Signature
public String contextsobjecttype {get; set;}

Property Value
Type: String

defaultvalues
The action's default values.
Signature
public List<QuickAction.DescribeQuickActionDefaultValue> defaultvalues {get; set;}

Property Value
Type: List<QuickAction.DescribeQuickActionDefaultValue>

flowdevname
If the custom action invokes a flow, the fully qualified name of the flow.

Signature
public String flowdevname {get; set;}

Property Value
Type: String

flowrecordidvar
If the custom action invokes a flow, the input variable that the custom action passes the record’s ID to.

Signature
public String flowrecordidvar {get; set;}

Property Value
Type: String
Valid values are null or recordId.

height
The height in pixels of the action pane.

Signature
public Integer height {get; set;}

Property Value
Type: Integer

iconname
The name of the icon used for the action. If a custom icon is not used, this value isn’t set.
Signature
public String iconname {get; set;}

Property Value
Type: String

icons
Array of icons. Each icon is associated with a theme.

Signature
public List<Schema.DescribeIconResult> icons {get; set;}

Property Value
Type: List<Schema.DescribeIconResult> on page 2730
If no custom icon was associated with the quick action and the quick action creates a specific object, the icons will correspond to the icons used for the created object. For example, if the quick action creates an Account, the icon array will contain the icons used for Account.
If a custom icon was associated with the quick action, the array will contain that custom icon.

iconurl
The URL of the icon used for the action. This icon URL corresponds to the 32x32 icon used for the current Salesforce theme, introduced in Spring '10, or the custom icon, if there is one.

Signature
public String iconurl {get; set;}

Property Value
Type: String

layout
The section of the layout where the action resides.

Signature
public QuickAction.DescribeLayoutSection layout {get; set;}

Property Value
Type: QuickAction.DescribeLayoutSection on page 2497
**lightningcomponentbundleid**
If the custom action invokes a Lightning component, the ID of the Lightning component bundle to which the component belongs.

**Signature**

```java
public String lightningcomponentbundleid {get; set;}
```

**Property Value**
Type: String

**lightningcomponentbundlename**
If the custom action invokes a Lightning component, the name of the Lightning component bundle to which the component belongs.

**Signature**

```java
public String lightningcomponentbundlename {get; set;}
```

**Property Value**
Type: String

**lightningcomponentqualifiedname**
The fully qualified name of the Lightning component invoked by the custom action.

**Signature**

```java
public String lightningcomponentqualifiedname {get; set;}
```

**Property Value**
Type: String

**miniiconurl**
The icon's URL. This icon URL corresponds to the 16x16 icon used for the current Salesforce theme, introduced in Spring '10, or the custom icon, if there is one.

**Signature**

```java
public String miniiconurl {get; set;}
```

**Property Value**
Type: String
**showquickactionlcheader**
Indicates whether the Lightning component quick action header and footer are shown. If `false`, then both the header containing the quick action title and the footer containing the Save and Cancel buttons aren’t displayed.

**Signature**
```
public Boolean showquickactionlcheader {get; set;}
```

**Property Value**
Type: Boolean

**showquickactionvfheader**
Indicates whether the Visualforce quick action header and footer should be shown. If `false`, then both the header containing the quick action title and the footer containing the Save and Cancel buttons aren’t displayed.

**Signature**
```
public Boolean showquickactionvfheader {get; set;}
```

**Property Value**
Type: Boolean

**targetparentfield**
The parent object type of the action. Links the target object to the parent object. For example, the value is Account if the target object is Contact and the parent object is Account.

**Signature**
```
public String targetparentfield {get; set;}
```

**Property Value**
Type: String

**targetrecordtypeid**
The record type of the target record.

**Signature**
```
public String targetrecordtypeid {get; set;}
```

**Property Value**
Type: String
**targetsobjecttype**
The action's target object type.

**Signature**
```
public String targetsobjecttype {get; set;}
```

**Property Value**
Type: String

**visualforcepagename**
The name of the Visualforce page associated with the custom action.

**Signature**
```
public String visualforcepagename {get; set;}
```

**Property Value**
Type: String

**visualforcepageurl**
The URL of the Visualforce page associated with the action.

**Signature**
```
public String visualforcepageurl {get; set;}
```

**Property Value**
Type: String

**width**
The width in pixels of the action pane, for custom actions that call Visualforce pages, Canvas apps, or Lightning components.

**Signature**
```
public Integer width {get; set;}
```

**Property Value**
Type: Integer

**DescribeQuickActionResult Methods**
The following are methods for DescribeQuickActionResult. All are instance methods.
IN THIS SECTION:

- **getActionEnumOrId()**
  Returns the unique ID for the action. If the action doesn't have an ID, its API name is used.

- **getCanvasApplicationName()**
  Returns the name of the Canvas application, if used.

- **getColors()**
  Returns an array of color information. Each color is associated with a theme.

- **getContextSobjectType()**
  Returns the object used for the action. Replaces `getsourceSobjectType()` in API version 30.0 and later.

- **getDefaultValues()**
  Returns the default values for a action.

- **getFlowDevName()**
  If the custom action invokes a flow, returns the fully qualified name of the flow invoked by the custom action.

- **getFlowRecordIdVar()**
  If the custom action invokes a flow, returns the input variable that the custom action passes the record’s ID to.

- **getHeight()**
  Returns the height in pixels of the action pane.

- **getIconName()**
  Returns the action’s icon name.

- **getIconUrl()**
  Returns the URL of the 32x32 icon used for the action.

- **getIcons()**
  Returns a list of `Schema.DescribeIconResult` objects that describe colors used in a tab.

- **getLabel()**
  Returns the action label.

- **getLayout()**
  Returns the layout sections that comprise an action.

- **getLightningComponentBundleId()**
  If the custom action invokes a Lightning component, returns the ID of the Lightning component bundle to which the component belongs.

- **getLightningComponentBundleName()**
  If the custom action invokes a Lightning component, returns the name of the Lightning component bundle to which the component belongs.

- **getLightningComponentQualifiedName()**
  If the custom action invokes a Lightning component, returns the fully qualified name of the Lightning component invoked by the custom action.

- **getMiniIconUrl()**
  Returns the 16x16 icon URL.

- **getName()**
  Returns the action name.
getShowQuickActionLcHeader()
Returns an indication of whether the Lightning component quick action header and footer are shown.

getShowQuickActionVfHeader()
Returns an indication of whether the Visualforce quick action header and footer should be shown.

ggetSourceSobjectType()
Returns the object type used for the action.

gGetTargetParentField()
Returns the parent object’s type for the action.

gGetTargetRecordTypeId()
Returns the record type of the targeted record.

gGetTargetSobjectType()
Returns the action’s target object type.

gGetType()
Returns a create or custom Visualforce action.

gGetVisualforcePageName()
If Visualforce is used, returns the name of the associated page for the action.

gGetVisualforcePageUrl()
Returns the URL of the Visualforce page associated with the action.

gGetWidth()
If a custom action is created, returns the width in pixels of the action pane.

gGetActionEnumOrId()
Returns the unique ID for the action. If the action doesn’t have an ID, its API name is used.

**Signature**

```java
public String getActionEnumOrId()
```

**Return Value**

Type: `String`

**getCanvasApplicationName()**

Returns the name of the Canvas application, if used.

**Syntax**

```java
public String getCanvasApplicationName()
```

**Return Value**

Type: `String`
getColors()
Returns an array of color information. Each color is associated with a theme.

Signature
public List<Schema.DescribeColorResult> getColors()

Return Value
Type: List<Schema.DescribeColorResult>

getContextSobjectType()
Returns the object used for the action. Replaces getSourceSobjectType() in API version 30.0 and later.

Signature
public String getContextSobjectType()

Return Value
Type: String

getDefaultValues()
Returns the default values for a action.

Signature
public List<QuickAction.DescribeQuickActionDefaultValue> getDefaultValues()

Return Value
Type: List<QuickAction.DescribeQuickActionDefaultValue>

getFlowDevName()
If the custom action invokes a flow, returns the fully qualified name of the flow invoked by the custom action.

Signature
public String getFlowDevName()

Return Value
Type: String

getFlowRecordIdVar()
If the custom action invokes a flow, returns the input variable that the custom action passes the record’s ID to.
Signature
public String getFlowRecordIdVar()

Return Value
Type: String

getHeight()
Returns the height in pixels of the action pane.

Signature
public Integer getHeight()

Return Value
Type: Integer

getIconName()
Returns the actions' icon name.

Signature
public String getIconName()

Return Value
Type: String

getIconUrl()
Returns the URL of the 32x32 icon used for the action.

Signature
public String getIconUrl()

Return Value
Type: String

getIcons()
Returns a list of Schema.DescribeIconResult objects that describe colors used in a tab.

Signature
public List<Schema.DescribeIconResult> getIcons()
Return Value
Type: List<Schema.DescribeIconResult>

getLabel()
Returns the action label.

Signature
public String getLabel()

Return Value
Type: String

getLayout()
Returns the layout sections that comprise an action.

Signature
public QuickAction.DescribeLayoutSection getLayout()

Return Value
Type: QuickAction.DescribeLayoutSection

g LightningComponentBundleId()
If the custom action invokes a Lightning component, returns the ID of the Lightning component bundle to which the component belongs.

Signature
public String getLightningComponentBundleId()

Return Value
Type: String

g LightningComponentBundleName()
If the custom action invokes a Lightning component, returns the name of the Lightning component bundle to which the component belongs.

Signature
public String getLightningComponentBundleName()
Return Value
Type: String

**getLightningComponentQualifiedName()**
If the custom action invokes a Lightning component, returns the fully qualified name of the Lightning component invoked by the custom action.

**Signature**
```java
public String getLightningComponentQualifiedName()
```

**Return Value**
Type: String

**getMiniIconUrl()**
Returns the 16x16 icon URL.

**Signature**
```java
public String getMiniIconUrl()
```

**Return Value**
Type: String

**getName()**
Returns the action name.

**Signature**
```java
public String getName()
```

**Return Value**
Type: String

**getShowQuickActionLcHeader()**
Returns an indication of whether the Lightning component quick action header and footer are shown.

**Signature**
```java
public Boolean getShowQuickActionLcHeader()
```
Return Value
Type: Boolean
If false, then both the header containing the quick action title and the footer containing the Save and Cancel buttons aren’t displayed.

getShowQuickActionVfHeader()
Returns an indication of whether the Visualforce quick action header and footer should be shown.

Signature
public Boolean getShowQuickActionVfHeader()

Return Value
Type: Boolean
If false, then both the header containing the quick action title and the footer containing the Save and Cancel buttons aren’t displayed.

getSourceSobjectType()
Returns the object type used for the action.

Signature
public String getSourceSobjectType()

Return Value
Type: String

getTargetParentField()
Returns the parent object’s type for the action.

Signature
public String getTargetParentField()

Return Value
Type: String

getTargetRecordTypeId()
Returns the record type of the targeted record.

Signature
public String getTargetRecordTypeId()
Return Value
Type: String

getTargetSobjectType() Returns the action’s target object type.

Signature
public String getTargetSobjectType()

Return Value
Type: String

getType() Returns a create or custom Visualforce action.

Signature
public String getType()

Return Value
Type: String

getVisualforcePageName() If Visualforce is used, returns the name of the associated page for the action.

Signature
public String getVisualforcePageName()

Return Value
Type: String

getVisualforcePageUrl() Returns the URL of the Visualforce page associated with the action.

Signature
public String getVisualforcePageUrl()

Return Value
Type: String
**getWidth()**
If a custom action is created, returns the width in pixels of the action pane.

**Signature**
```java
public Integer getWidth()
```

**Return Value**
Type: Integer

### QuickActionDefaults Class
Represents an abstract Apex class that provides the context for running the standard Email Action on Case Feed and the container of the Email Message fields for the action payload. You can override the target fields before the standard Email Action is rendered.

**Namespace**
QuickAction

**Usage**

> **Note:** You cannot extend this abstract class. You can use the getter methods when using it in the context of QuickAction.QuickActionDefaultsHandler. Salesforce provides a class that extends this class (See QuickAction.SendEmailQuickActionDefaults.)

IN THIS SECTION:
QuickActionDefaults Methods

### QuickActionDefaults Methods
The following are methods for QuickActionDefaults.

IN THIS SECTION:
- `getActionName()`
- `getActionType()`
- `getContextId()`
- `getTargetSObject()`
getActionName()
Returns the name of the standard Email Action on Case Feed (Case.Email).

Signature
public String getActionName()

Return Value
Type: String

getActionType()
Returns the type of the standard Email Action on Case Feed (Email).

Signature
public String getActionType()

Return Value
Type: String

getContextId()
The ID of the context related to the standard Email Action on Case Feed (Case ID).

Signature
public Id getContextId()

Return Value
Type: Id

getTargetSObject()
The target object of the standard Email Action on Case Feed (EmailMessage).

Signature
public SObject getTargetSObject()

Return Value
Type: SObject
QuickActionDefaultsHandler Interface

The QuickAction.QuickActionDefaultsHandler interface lets you specify the default values for the standard Email and Send Email actions in the case feed. You can use this interface to specify the From address, CC address, BCC address, subject, and email body for the Email action in the case feed. You can use the interface to pre-populate these fields based on the context where the action is displayed, such as the case origin (for example, country) and subject.

Namespace

QuickAction

Usage

To specify default values for the standard Email action in the case feed, create a class that implements QuickAction.QuickActionDefaultsHandler.

The QuickAction.QuickActionDefaultsHandler interface works in Salesforce Classic and Lightning Experience.

When working in Lightning Experience, keep the following things in mind:

• The interface overrides email values set up with predefined IDs.
• The interface works with the out-of-the-box Email action provided on cases. You can also use the interface with custom Email actions for the case object.
• The interface in Lightning Experience doesn’t support:
  – Email attachments
  – Custom email fields
  – Visualforce email templates, which are a type of email template available in Salesforce Classic
• The From field determines the from address picklist. While you can’t customize this picklist in Send Email action types via the QuickActionDefaultsHandler interface, you can customize the From Address field. To customize this field, remove the From field from the SendEmail quick action layout and add the From Address field instead. Then provide a valid and verified from address in the QuickActionDefaultsHandler code. This address must be the current user’s address, an organization-wide email address that the current user has access to, or an Email-to-Case routing address.
• If your Apex interface adds content to the email body, merge fields display as unresolved. During preview and send, the merge fields resolve.

When you implement this interface, provide an empty parameterless constructor.

IN THIS SECTION:

QuickActionDefaultsHandler Methods
QuickActionDefaultsHandler Example Implementations

QuickActionDefaultsHandler Methods

The following are methods for QuickActionDefaultsHandler.
IN THIS SECTION:

**onInitDefaults(actionDefaults)**
Implement this method to provide default values for the standard Email action in the case feed.

**Signature**

```java
public void onInitDefaults(QuickAction.QuickActionDefaults[] actionDefaults)
```

**Parameters**

- `actionDefaults`
  Type: `QuickAction.QuickActionDefaults[]`
  This array contains only one item of type `QuickAction.SendEmailQuickActionDefaults`.

**Return Value**

Type: void

**QuickActionDefaultsHandler Example Implementations**

These examples are implementations of the `QuickAction.QuickActionDefaultsHandler` interface.

In this example, the `onInitDefaults` method checks whether the element passed in the array is for the standard Email action in the case feed. Then, it performs a query to retrieve the case that corresponds to the context ID. Next, it sets the value of the BCC address of the corresponding email message to a default value. The default value is based on the case reason. Finally, it sets the default values of the email template properties. The `onInitDefaults` method determines the default values based on two criteria: first, whether a reply action on an email message initiated the call to the method, and second, whether any previous emails attached to the case are associated with the call.

```java
global class EmailPublisherLoader implements QuickAction.QuickActionDefaultsHandler {
    // Empty constructor
    global EmailPublisherLoader() {
    }

    // The main interface method
    global void onInitDefaults(QuickAction.QuickActionDefaults[] defaults) {
        QuickAction.SendEmailQuickActionDefaults sendEmailDefaults = null;

        // Check if the quick action is the standard case feed Send Email action
        for (Integer j = 0; j < defaults.size(); j++) {
            if (defaults.get(j) instanceof QuickAction.SendEmailQuickActionDefaults &&
                defaults.get(j).getTargetSObject().getSObjectType() == EmailMessage.sObjectType &&
                defaults.get(j).getActionName().equals('Case.Email') &&
                defaults.get(j).getActionType().equals('Email')) {
                sendEmailDefaults =
            }
        }
    }
}
```
if (sendEmailDefaults != null) {
    Case c = [SELECT Status, Reason FROM Case
                WHERE Id=:sendEmailDefaults.getContextId()];
    EmailMessage emailMessage = (EmailMessage)sendEmailDefaults.getTargetSObject();
    // Set BCC address to make sure each email goes for audit
    emailMessage.BccAddress = getBccAddress(c.Reason);
    /*
    Set Template related fields
    when the In Reply To Id field is null we know the interface
    is called on page load. Here we check if
    there are any previous emails attached to the case and load
    the 'New_Case_Created' or 'Automatic_Response' template.
    When the In Reply To Id field is not null we know that
    the interface is called on click of reply/reply all
    of an email and we load the 'Default_reply_template' template */
    if (sendEmailDefaults.getInReplyToId() == null) {
        Integer emailCount = [SELECT count() FROM EmailMessage
                               WHERE ParentId=:sendEmailDefaults.getContextId()];
        if (emailCount != null && emailCount > 0) {
            sendEmailDefaults.setTemplateId(
                getTemplateIdHelper('Automatic_Response'));
        } else {
            sendEmailDefaults.setTemplateId(
                getTemplateIdHelper('New_Case_Created'));
        }
        sendEmailDefaults.setInsertTemplateBody(false);
        sendEmailDefaults.setIgnoreTemplateSubject(false);
    } else {
        sendEmailDefaults.setTemplateId(
            getTemplateIdHelper('Default_reply_template'));
        sendEmailDefaults.setInsertTemplateBody(false);
        sendEmailDefaults.setIgnoreTemplateSubject(true);
    }
}

private Id getTemplateIdHelper(String templateApiName) {
    Id templateId = null;
    try {
        templateId = [select id, name from EmailTemplate
                      where developername = :templateApiName].id;
    } catch (Exception e) {
        system.debug('Unable to locate EmailTemplate using name: ' +
                     templateApiName + ' refer to Setup | Communications Templates '
                     + templateApiName);
    }
}
private String getBccAddress(String reason) {
    if (reason != null && reason.equals('Technical')) {
        return 'support_technical@mycompany.com';
    } else if (reason != null && reason.equals('Billing')) {
        return 'support_billing@mycompany.com';
    } else {
        return 'support@mycompany.com';
    }
}

In this example, the `onInitDefaults` method checks whether the element passed in the array is for the standard Email action in the case feed. Then it performs a query to determine if the case Priority is set to `High`. If the Priority is set to `High`, the email address `managers@acme.com` is appended to the BCC field.

```java
global class EmailPublisherForHighPriorityCases implements QuickAction.QuickActionDefaultsHandler {
    // Empty constructor
    global EmailPublisherForHighPriorityCases() {
    }

    // The main interface method
    global void onInitDefaults(QuickAction.QuickActionDefaults[] defaults) {
        QuickAction.SendEmailQuickActionDefaults sendEmailDefaults = (QuickAction.SendEmailQuickActionDefaults)defaults.get(0);
        EmailMessage emailMessage = (EmailMessage)sendEmailDefaults.getTargetSObject();

        Case c = [SELECT CaseNumber, Priority FROM Case WHERE Id=:sendEmailDefaults.getContextId()];

        // If case severity is "High," append "managers@acme.com" to the existing (and possibly blank) BCC field
        if (c.Priority != null && c.Priority.equals('High')) {
            emailMessage.BccAddress = 'managers@acme.com';
        }
    }
}
```

In this example, the `onInitDefaults` method checks whether the element passed in the array is for the standard Email action in the case feed. Then it performs a query to determine if the case Type is set to `Problem`. If the type is set to `Problem`, the `First Response` email template is inserted into the body of the email.

```java
global class EmailPublisherForCaseType implements QuickAction.QuickActionDefaultsHandler {
    // Empty constructor
    global EmailPublisherForCaseType() {
    }

    // The main interface method
    global void onInitDefaults(QuickAction.QuickActionDefaults[] defaults) {
        QuickAction.SendEmailQuickActionDefaults sendEmailDefaults = (QuickAction.SendEmailQuickActionDefaults)defaults.get(0);
```

2523
In this example, the `onInitDefaults` method checks whether the element passed in the array is for the standard Email action in the case feed. Then it performs a query to determine if the email is a Reply or Reply All email. If email is a Reply or Reply All email, the corresponding email templates for these emails are inserted into the body of the email.

```apex
global class EmailPublisherForReplyAndReplyAll implements QuickAction.QuickActionDefaultsHandler {
    // Empty constructor
    global EmailPublisherForReplyAndReplyAll() {
    }

    // The main interface method
    global void onInitDefaults(QuickAction.QuickActionDefaults[] defaults) {
        QuickAction.SendEmailQuickActionDefaults sendEmailDefaults =
            (QuickAction.SendEmailQuickActionDefaults)defaults.get(0);
        EmailMessage emailMessage = (EmailMessage)sendEmailDefaults.getTargetSObject();

        // If the email is a “Reply” email, insert the “Reply Email Template” to the email body
        if (sendEmailDefaults.getActionName().equals('EmailMessage._Reply')) {
            sendEmailDefaults.setTemplateId('Insert Reply Email Template ID Here');
            sendEmailDefaults.setInsertTemplateBody(true);
            sendEmailDefaults.setIgnoreTemplateSubject(false);
        }

        // If the email is a “Reply All” email, insert the “Reply All Email Template” to the email body
        else if (sendEmailDefaults.getActionName().equals('EmailMessage._ReplyAll')) {
            sendEmailDefaults.setTemplateId('Insert Reply All Email Template ID Here');
            sendEmailDefaults.setInsertTemplateBody(true);
            sendEmailDefaults.setIgnoreTemplateSubject(false);
        }
    }
}
```
QuickActionRequest Class

Use the `QuickAction.QuickActionRequest` class for providing action information for quick actions to be performed by `QuickAction` class methods. Action information includes the action name, context record ID, and record.

Namespace

`QuickAction`

Usage

For Apex saved using Salesforce API version 28.0, a parent ID is associated with the `QuickActionRequest` instead of the context ID. The constructor of this class takes no arguments:

```java
QuickAction.QuickActionRequest qar = new QuickAction.QuickActionRequest();
```

Example

In this sample, a new quick action is created to create a contact and assign a record to it.

```java
QuickAction.QuickActionRequest req = new QuickAction.QuickActionRequest();
// Some quick action name
req.quickActionName = Schema.Account.QuickAction.AccountCreateContact;

// Define a record for the quick action to create
Contact c = new Contact();
c.lastname = 'last name';
req.record = c;

// Provide the context ID (or parent ID). In this case, it is an Account record.
req.contextid = '001xx000003DGcO';

QuickAction.QuickActionResult res = QuickAction.performQuickAction(req);
```

SEE ALSO:

- `QuickAction Class`

QuickActionRequest Constructors

The following are constructors for `QuickActionRequest`. 
QuickActionRequest()  
Creates a new instance of the `QuickAction.QuickActionRequest` class.

**Signature**  
```
public QuickActionRequest()
```

**QuickActionRequest Methods**  
The following are methods for `QuickActionRequest`. All are instance methods.

**getContextId()**  
Returns this QuickAction’s context record ID.

**getQuickActionName()**  
Returns this QuickAction’s name.

**getRecord()**  
Returns the QuickAction’s associated record.

**setContextId(contextId)**  
Sets this QuickAction’s context ID. Returned by `getContextId`.

**setQuickActionName(name)**  
Sets this QuickAction’s name. Returned by `getQuickActionName`.

**setRecord(record)**  
Sets a record for this QuickAction. Returned by `getRecord`.

**getContextId()**  
Returns this QuickAction’s context record ID.

**Signature**  
```
public Id getContextId()
```

**Return Value**  
Type: ID

**getQuickActionName()**  
Returns this QuickAction’s name.
Signature

```java
public String getQuickActionName()
```

Return Value
Type: `String`

```java
getRecord()
```

Returns the QuickAction’s associated record.

Signature

```java
public SObject getRecord()
```

Return Value
Type: `SObject`

```java
setContextId(contextId)
```

Sets this QuickAction’s context ID. Returned by `getContextId`.

Signature

```java
public Void setContextId(Id contextId)
```

Parameters
```
contextId
Type: `Id`
```

Return Value
Type: `Void`

Usage
For Apex saved using Salesforce API version 28.0, sets this QuickAction’s parent ID and is returned by `getParentId`.

```java
setQuickActionName(name)
```

Sets this QuickAction’s name. Returned by `getQuickActionName`.

Signature

```java
public Void setQuickActionName(String name)
```
Parameters
name
  Type: String

Return Value
Type: Void

setRecord(record)
Sets a record for this QuickAction. Returned by getRecord.

Signature
public Void setRecord(SObject record)

Parameters
record
  Type: SObject

Return Value
Type: Void

QuickActionResult Class
After you initiate a quick action with the QuickAction class, use the QuickActionResult class for processing action results.

Namespace
QuickAction

SEE ALSO:
  QuickAction Class

QuickActionResult Methods
The following are methods for QuickActionResult. All are instance methods.

IN THIS SECTION:
  getErrors()
  If an error occurs, an array of one or more database error objects, along with error codes and descriptions, is returned.
  getIds()
  The IDs of the QuickActions being processed.
  getSuccessMessage()
  Returns the success message associated with the quick action.
isCreated()
Returns `true` if the action is created; otherwise, `false`.

isSuccess()
Returns `true` if the action completes successfully; otherwise, `false`.

getErrors()
If an error occurs, an array of one or more database error objects, along with error codes and descriptions, is returned.

Signature
```
public List<Database.Error> getErrors()
```

Return Value
Type: `List<Database.Error>`

getIds()
The IDs of the QuickActions being processed.

Signature
```
public List<Id> getIds()
```

Return Value
Type: `List<Id>`

getSuccessMessage()
Returns the success message associated with the quick action.

Signature
```
public String getSuccessMessage()
```

Return Value
Type: `String`

isCreated()
Returns `true` if the action is created; otherwise, `false`.

Signature
```
public Boolean isCreated()
```
Return Value
Type: Boolean

**isSuccess()**
Returns `true` if the action completes successfully; otherwise, `false`.

Signature
```
public Boolean isSuccess()
```

Return Value
Type: Boolean

**SendEmailQuickActionDefaults Class**

Represents an Apex class that provides: the From address list; the original email's email message ID, provided that the reply action was invoked on the email message feed item; and methods to specify related settings on templates. You can override these fields before the standard Email Action is rendered.

Namespace
QuickAction

Usage

**Note:** You cannot instantiate this class. One can use the getters/setters when using it in the context of `QuickAction.QuickActionDefaultsHandler`.

IN THIS SECTION:

SendEmailQuickActionDefaults Methods

**SendEmailQuickActionDefaults Methods**
The following are methods for `SendEmailQuickActionDefaults`.

IN THIS SECTION:

getFromAddressList()
Returns a list of email addresses that are available in the From: address drop-down menu for the standard Email Action.

getInReplyTold()
Returns the email message ID of the email to which the reply/reply all action has been invoked.

setIgnoreTemplateSubject(useOriginalSubject)
Specifies whether the template subject should be ignored (true), thus using the original subject, or whether the template subject should replace the original subject (false).
setInsertTemplateBody(keepOriginalBodyContent)
Specifies whether the template body should be inserted above the original body content (true) or whether it should replace the entire content with the template body (false).

setTemplateId(templatedId)
Sets the email template ID to load into the email body.

getFromAddressList()
Returns a list of email addresses that are available in the From: address drop-down menu for the standard Email Action.

Signature
public List<String> getFromAddressList()

Return Value
Type: List<String>

getInReplyToId()
Returns the email message ID of the email to which the reply/reply all action has been invoked.

Signature
public Id getInReplyToId()

Return Value
Type: Id

setIgnoreTemplateSubject(useOriginalSubject)
Specifies whether the template subject should be ignored (true), thus using the original subject, or whether the template subject should replace the original subject (false).

Signature
public void setIgnoreTemplateSubject(Boolean useOriginalSubject)

Parameters
useOriginalSubject
Type: Boolean

Return Value
Type: void
**setInsertTemplateBody(keepOriginalBodyContent)**

Specifies whether the template body should be inserted above the original body content (true) or whether it should replace the entire content with the template body (false).

**Signature**

`public void setInsertTemplateBody(Boolean keepOriginalBodyContent)`

**Parameters**

`keepOriginalBodyContent`

Type: Boolean

**Return Value**

Type: void

**setTemplateId(templateId)**

Sets the email template ID to load into the email body.

**Signature**

`public void setTemplateId(Id templateId)`

**Parameters**

`templateId`

Type: Id

The template ID.

**Return Value**

Type: void

**Reports Namespace**

The Reports namespace provides classes for accessing the same data as is available in the Salesforce Reports and Dashboards REST API.

The following are the classes in the Reports namespace.

**IN THIS SECTION:**

- **AggregateColumn Class**
  
  Contains methods for describing summary fields such as Record Count, Sum, Average, Max, Min, and custom summary formulas. Includes name, label, data type, and grouping context.
BucketField Class
Contains methods and constructors to work with information about a bucket field, including bucket type, name, and bucketed values.

BucketFieldValue Class
Contains information about the report values included in a bucket field.

BucketType Enum
The types of values included in a bucket.

ColumnDataType Enum
The Reports.ColumnDataType enum describes the type of data in a column. It is returned by the getDataType method.

ColumnSortOrder Enum
The Reports.ColumnSortOrder enum describes the order that the grouping column uses to sort data.

CrossFilter Class
Contains methods and constructors used to work with information about a cross filter.

CsfGroupType Enum
The group level at which the custom summary format aggregate is displayed in a report.

DateGranularity Enum
The Reports.DateGranularity enum describes the date interval that is used for grouping.

DetailColumn Class
 Contains methods for describing fields that contain detailed data. Detailed data fields are also listed in the report metadata.

Dimension Class
Contains information for each row or column grouping.

EvaluatedCondition Class
Contains the individual components of an evaluated condition for a report notification, such as the aggregate name and label, the operator, and the value that the aggregate is compared to.

EvaluatedConditionOperator Enum
The Reports.EvaluatedConditionOperator enum describes the type of operator used to compare an aggregate to a value. It is returned by the getOperator method.

FilterOperator Class
Contains information about a filter operator, such as display name and API name.

FilterValue Class
Contains information about a filter value, such as the display name and API name.

FormulaType Enum
The format of the numbers in a custom summary formula.

GroupingColumn Class
Contains methods for describing fields that are used for column grouping.

GroupingInfo Class
Contains methods for describing fields that are used for grouping.

GroupingValue Class
Contains grouping values for a row or column, including the key, label, and value.

NotificationAction Interface
Implement this interface to trigger a custom Apex class when the conditions for a report notification are met.
NotificationActionContext Class
Contains information about the report instance and condition threshold for a report notification.

ReportCsf Class
Contains methods and constructors for working with information about a custom summary formula (CSF).

ReportCurrency Class
Contains information about a currency value, including the amount and currency code.

ReportDataCell Class
Contains the data for a cell in the report, including the display label and value.

ReportDescribeResult Class
Contains report, report type, and extended metadata for a tabular, summary, or matrix report.

ReportDetailRow Class
Contains data cells for a detail row of a report.

ReportDivisionInfo Class
Contains information about the divisions that can be used to filter a report.

ReportExtendedMetadata Class
Contains report extended metadata for a tabular, summary, or matrix report.

ReportFact Class
Contains the fact map for the report, which represents the report’s data values.

ReportFactWithDetails Class
Contains the detailed fact map for the report, which represents the report’s data values.

ReportFactWithSummaries Class
Contains the fact map for the report, which represents the report’s data values, and includes summarized fields.

ReportFilter Class
Contains information about a report filter, including column, operator, and value.

ReportFormat Enum
Contains the possible report format types.

ReportFilterType Enum
The types of values included in a report filter type.

ReportInstance Class
Returns an instance of a report that was run asynchronously. Retrieves the results for that instance.

ReportManager Class
Runs a report synchronously or asynchronously and with or without details.

ReportMetadata Class
Contains report metadata for a tabular, summary, or matrix report.

ReportResults Class
Contains the results of running a report.

ReportScopeInfo Class
Contains information about possible scope values that you can choose. Scope values depend on the report type. For example, you can set the scope for opportunity reports to All opportunities, My team’s opportunities, or My opportunities.
ReportScopeValue Class
Contains information about a possible scope value. Scope values depend on the report type. For example, you can set the scope for opportunity reports to All opportunities, My team’s opportunities, or My opportunities.

ReportType Class
Contains the unique API name and display name for the report type.

ReportTypeColumn Class
Contains detailed report type metadata about a field, including data type, display name, and filter values.

ReportTypeColumnCategory Class
Information about categories of fields in a report type.

ReportTypeMetadata Class
Contains report type metadata, which gives you information about the fields that are available in each section of the report type, plus filter information for those fields.

SortColumn Class
Contains information about the sort column used in the report.

StandardDateFilter Class
Contains information about standard date filter available in the report—for example, the API name, start date, and end date of the standard date filter duration as well as the API name of the date field on which the filter is placed.

StandardDateFilterDuration Class
Contains information about each standard date filter—also referred to as a relative date filter. It contains the API name and display label of the standard date filter duration as well as the start and end dates.

StandardDateFilterDurationGroup Class
Contains information about the standard date filter groupings, such as the grouping display label and all standard date filters that fall under the grouping. Groupings include Calendar Year, Calendar Quarter, Calendar Month, Calendar Week, Fiscal Year, Fiscal Quarter, Day, and custom values based on user-defined date ranges.

StandardFilter Class
Contains information about the standard filter defined in the report, such as the filter field API name and filter value.

StandardFilterInfo Class
Is an abstract base class for an object that provides standard filter information.

StandardFilterInfoPicklist Class
Contains information about the standard filter picklist, such as the display name and type of the filter field, the default picklist value, and a list of all possible picklist values.

StandardFilterType Enum
The StandardFilterType enum describes the type of standard filters in a report. The getType() method returns a Reports.StandardFilterType enum value.

SummaryValue Class
Contains summary data for a cell of the report.

ThresholdInformation Class
Contains a list of evaluated conditions for a report notification.

TopRows Class
Contains methods and constructors for working with information about a row limit filter.

Reports Exceptions
The Reports namespace contains exception classes.
AggregateColumn Class

Contains methods for describing summary fields such as Record Count, Sum, Average, Max, Min, and custom summary formulas. Includes name, label, data type, and grouping context.

Namespace

Reports

AggregateColumn Methods

The following are methods for AggregateColumn. All are instance methods.

IN THIS SECTION:

getName()
Returns the unique API name of the summary field.

getLabel()
Returns the localized display name for the summarized or custom summary formula field.

dataType()
Returns the data type of the summarized or custom summary formula field.

getAcrossGroupingContext()
Returns the column grouping in the report where the summary field is displayed.

downGroupingContext()
Returns the row grouping in the report where the summary field is displayed.

tName()
Returns the unique API name of the summary field.

Syntax

public String getName()

Return Value

Type: String

tLabel()
Returns the localized display name for the summarized or custom summary formula field.

Syntax

public String getLabel()
**Return Value**
Type: String

**getDataType()**
Returns the data type of the summarized or custom summary formula field.

**Syntax**
```java
public String getDataType()
```

**getAcrossGroupingContext()**
Returns the column grouping in the report where the summary field is displayed.

**Syntax**
```java
public String getAcrossGroupingContext()
```

**getDownGroupingContext()**
Returns the row grouping in the report where the summary field is displayed.

**Syntax**
```java
public String getDownGroupingContext()
```

**BucketField Class**
Contains methods and constructors to work with information about a bucket field, including bucket type, name, and bucketed values.

**Namespace**
Reports

IN THIS SECTION:
BucketField Constructors
BucketField Methods

BucketField Constructors
The following are constructors for BucketField.

IN THIS SECTION:

BucketField(bucketType, devloperName, label, nullTreatedAsZero, otherBucketLabel, sourceColumnName, values)
Creates an instance of the Reports.BucketField class using the specified parameters.

BucketField()
Creates an instance of the Reports.BucketField class. You can then set values by using the class’s set methods.

**BucketField(bucketType, devloperName, label, nullTreatedAsZero, otherBucketLabel, sourceColumnName, values)**
Creates an instance of the Reports.BucketField class using the specified parameters.

**Signature**

```
public BucketField(Reports.BucketType bucketType, String devloperName, String label,
                   Boolean nullTreatedAsZero, String otherBucketLabel, String sourceColumnName,
                   List<Reports.BucketFieldValue> values)
```

**Parameters**

- **bucketType**
  Type: `Reports.BucketType`
  The type of bucket.

- **devloperName**
  Type: `String`
  API name of the bucket.

- **label**
  Type: `String`
  User-facing name of the bucket.

- **nullTreatedAsZero**
  Type: `Boolean`
  Specifies whether null values are converted to zero (`true`) or not (`false`).

- **otherBucketLabel**
  Type: `String`
  Name of the fields grouped as Other (in buckets of BucketType PICKLIST).

- **sourceColumnName**
  Type: `String`
  Name of the bucketed field.
values
Type: List<Reports.BucketType>
Types of the values included in the bucket.

BucketField()
Creates an instance of the Reports.BucketField class. You can then set values by using the class’s set methods.

Signature
public BucketField()

BucketField Methods
The following are methods for BucketField.

IN THIS SECTION:
getBucketType()
Returns the bucket type.
getDevloperName()
Returns the bucket’s API name.
setLabel()
Returns the user-facing name of the bucket.
getNullTreatedAsZero()
Returns true if null values are converted to the number zero, otherwise returns false.
getOtherBucketLabel()
Returns the name of fields grouped as Other in buckets of type PICKLIST.
getSourceColumnName()
Returns the API name of the bucketed field.
getValues()
Returns the report values grouped by the bucket field.
setBucketType(value)
Sets the BucketType of the bucket.
setBucketType(bucketType)
Sets the BucketType of the bucket.
setDevloperName(devloperName)
Sets the API name of the bucket.
setLabel(label)
Sets the user-facing name of the bucket.
setNullTreatedAsZero(nullTreatedAsZero)
Specifies whether null values in the bucket are converted to zero (true) or not (false).
setOtherBucketLabel(otherBucketLabel)
Sets the name of the fields grouped as Other (in buckets of BucketType PICKLIST).

setSourceColumnName(sourceColumnName)
Specifies the name of the bucketed field.

setValues(values)
Specifies which type of values are included in the bucket.

toString()
Returns a string.

getBucketType()
Returns the bucket type.

Signature
public Reports.BucketType getBucketType()

Return Value
Type: Reports.BucketType

getDevloperName()
Returns the bucket’s API name.

Signature
public String getDevloperName()

Return Value
Type: String

getLabel()
Returns the user-facing name of the bucket.

Signature
public String getLabel()

Return Value
Type: String

getNullTreatedAsZero()
Returns true if null values are converted to the number zero, otherwise returns false.
getNullTreatedAsZero()  
Returns whether null values are treated as zero.

Signature  
```java
public Boolean getNullTreatedAsZero()
```

getOtherBucketLabel()  
Returns the name of fields grouped as `Other` in buckets of type `PICKLIST`.

Signature  
```java
public String getOtherBucketLabel()
```

getSourceColumnName()  
Returns the API name of the bucketed field.

Signature  
```java
public String getSourceColumnName()
```

getValues()  
Returns the report values grouped by the bucket field.

Signature  
```java
public List<Reports.BucketFieldValue> getValues()
```

setBucketType(value)  
Sets the `BucketType` of the bucket.

Signature  
```java
public void setBucketType(String value)
```
Parameters

value
  Type: String
  See the Reports.BucketType enum for valid values.

Return Value

Type: void

setBucketType (bucketType)
Sets the BucketType of the bucket.

Signature

public void setBucketType(Reports.BucketType bucketType)

Parameters

bucketType
  Type: Reports.BucketType

Return Value

Type: void

setDevloperName (devloperName)
Sets the API name of the bucket.

Signature

public void setDevloperName(String devloperName)

Parameters

devloperName
  Type: String
  The API name to assign to the bucket.

Return Value

Type: void

setLabel (label)
Sets the user-facing name of the bucket.
Signature
public void setLabel(String label)

Parameters
label
Type: String

Return Value
Type: void

`setNullTreatedAsZero(nullTreatedAsZero)`
Specifies whether null values in the bucket are converted to zero (true) or not (false).

Signature
public void setNullTreatedAsZero(Boolean nullTreatedAsZero)

Parameters
nullTreatedAsZero
Type: Boolean

Return Value
Type: void

`setOtherBucketLabel(otherBucketLabel)`
Sets the name of the fields grouped as Other (in buckets of BucketType PICKLIST).

Signature
public void setOtherBucketLabel(String otherBucketLabel)

Parameters
otherBucketLabel
Type: String

Return Value
Type: void

`setSourceColumnName(sourceColumnName)`
Specifies the name of the bucketed field.
Signature

```java
public void setSourceColumnName(String sourceColumnName)
```

Parameters

`sourceColumnName`
Type: String

Return Value

Type: void

### setValues(values)

Specifies which type of values are included in the bucket.

Signature

```java
public void setValues(List<Reports.BucketFieldValue> values)
```

Parameters

`values`
Type: List on page 3177<Reports.BucketFieldValue>

Return Value

Type: void

### toString()

Returns a string.

Signature

```java
public String toString()
```

Return Value

Type: String

**BucketFieldValue Class**

Contains information about the report values included in a bucket.

**Namespace**

Reports
IN THIS SECTION:

BucketFieldValue Constructors

BucketFieldValue Methods

BucketFieldValue Constructors

The following are constructors for BucketFieldValue.

IN THIS SECTION:

BucketFieldValue(label, sourceDimensionValues, rangeUpperBound)
Creates an instance of the Reports.BucketFieldValue class using the specified parameters.

BucketFieldValue()
Creates an instance of the Reports.BucketFieldValue class. You can then set values by using the class’s set methods.

**BucketFieldValue(label, sourceDimensionValues, rangeUpperBound)**

Creates an instance of the Reports.BucketFieldValue class using the specified parameters.

**Signature**

```java
public BucketFieldValue(String label, List<String> sourceDimensionValues, Double rangeUpperBound)
```

**Parameters**

**label**
Type: **String**
The user-facing name of the bucket.

**sourceDimensionValues**
Type: **List on page 3177<String>**
A list of the values from the source field included in this bucket category (in buckets of type PICKLIST and buckets of type TEXT).

**rangeUpperBound**
Type: **Double**
The greatest range limit under which values are included in this bucket category (in buckets of type NUMBER).

**BucketFieldValue()**

Creates an instance of the Reports.BucketFieldValue class. You can then set values by using the class’s set methods.

**Signature**

```java
public BucketFieldValue()
```

**BucketFieldValue Methods**

The following are methods for BucketFieldValue.
IN THIS SECTION:

getLabel()
Returns the user-facing name of the bucket category.

getRangeUpperBound()
Returns the greatest range limit under which values are included in this bucket category (in buckets of type NUMBER).

getSourceDimensionValues()
Returns a list of the values from the source field included in this bucket category (in buckets of type PICKLIST and buckets of type TEXT).

setLabel(label)
Set the user-facing name of the bucket category.

setRangeUpperBound(rangeUpperBound)
Sets the greatest limit of a range under which values are included in this bucket category (in buckets of type NUMBER).

setSourceDimensionValues(sourceDimensionValues)
Specifies the values from the source field included in this bucket category (in buckets of type PICKLIST and buckets of type TEXT).

toString()
Returns a string.

getLabel()
Returns the user-facing name of the bucket category.

Signature

public String getLabel()

Return Value

Type: String

getRangeUpperBound()
Returns the greatest range limit under which values are included in this bucket category (in buckets of type NUMBER).

Signature

public Double getRangeUpperBound()

Return Value

Type: Double

getSourceDimensionValues()
Returns a list of the values from the source field included in this bucket category (in buckets of type PICKLIST and buckets of type TEXT).
**Signature**

```java
public List<String> getSourceDimensionValues()
```

**Return Value**

Type: `List<String>`

**setLabel(label)**

Set the user-facing name of the bucket category.

**Signature**

```java
public void setLabel(String label)
```

**Parameters**

- `label`
  Type: `String`

**Return Value**

Type: `void`

**setRangeUpperBound(rangeUpperBound)**

Sets the greatest limit of a range under which values are included in this bucket category (in buckets of type `NUMBER`).

**Signature**

```java
public void setRangeUpperBound(Double rangeUpperBound)
```

**Parameters**

- `rangeUpperBound`
  Type: `Double`

**Return Value**

Type: `void`

**setSourceDimensionValues(sourceDimensionValues)**

Specifies the values from the source field included in this bucket category (in buckets of type `PICKLIST` and buckets of type `TEXT`).

**Signature**

```java
public void setSourceDimensionValues(List<String> sourceDimensionValues)
```
Parameters

sourceDimensionValues
Type: List<String>

Return Value
Type: void

toString()
Returns a string.

Signature
public String toString()

Return Value
Type: String

BucketType Enum
The types of values included in a bucket.

Enum Values
The following are the values of the Reports.BucketType enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUMBER</td>
<td>Numeric values</td>
</tr>
<tr>
<td>PICKLIST</td>
<td>Picklist values</td>
</tr>
<tr>
<td>TEXT</td>
<td>String values</td>
</tr>
</tbody>
</table>

ColumnDataType Enum
The Reports.ColumnDataType enum describes the type of data in a column. It is returned by the getDataType method.

Namespace
Reports

Enum Values
The following are the values of the Reports.ColumnDataType enum.
<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOOLEAN_DATA</td>
<td>Boolean (true or false) values</td>
</tr>
<tr>
<td>COMBOBOX_DATA</td>
<td>Comboboxes, which provide a set of enumerated values and enable the user to specify a value that is not in the list</td>
</tr>
<tr>
<td>CURRENCY_DATA</td>
<td>Currency values</td>
</tr>
<tr>
<td>DATETIME_DATA</td>
<td>DateTime values</td>
</tr>
<tr>
<td>DATE_DATA</td>
<td>Date values</td>
</tr>
<tr>
<td>DOUBLE_DATA</td>
<td>Double values</td>
</tr>
<tr>
<td>EMAIL_DATA</td>
<td>Email addresses</td>
</tr>
<tr>
<td>ID_DATA</td>
<td>An object’s Salesforce ID</td>
</tr>
<tr>
<td>INT_DATA</td>
<td>Integer values</td>
</tr>
<tr>
<td>MULTIPICKLIST_DATA</td>
<td>Multi-select picklists, which provide a set of enumerated values from which multiple values can be selected</td>
</tr>
<tr>
<td>PERCENT_DATA</td>
<td>Percent values</td>
</tr>
<tr>
<td>PHONE_DATA</td>
<td>Phone numbers. Values can include alphabetic characters. Client applications are responsible for phone number formatting.</td>
</tr>
<tr>
<td>PICKLIST_DATA</td>
<td>Single-select picklists, which provide a set of enumerated values from which only one value can be selected</td>
</tr>
<tr>
<td>REFERENCE_DATA</td>
<td>Cross-references to another object, analogous to a foreign key field</td>
</tr>
<tr>
<td>STRING_DATA</td>
<td>String values</td>
</tr>
<tr>
<td>TEXTAREA_DATA</td>
<td>String values that are displayed as multiline text fields</td>
</tr>
<tr>
<td>TIME_DATA</td>
<td>Time values</td>
</tr>
<tr>
<td>URL_DATA</td>
<td>URL values that are displayed as hyperlinks</td>
</tr>
</tbody>
</table>

**ColumnSortOrder Enum**

The `Reports.ColumnSortOrder` enum describes the order that the grouping column uses to sort data.

**Namespace**

`Reports`

**Usage**

The `GroupingInfo.getColumnSortOrder()` method returns a `Reports.ColumnSortOrder` enum value. The `GroupingInfo.setColumnSortOrder()` method takes the enum value as an argument.
Enum Values

The following are the values of the `Reports.ColumnSortOrder` enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASCENDING</td>
<td>Sort data in ascending order (A–Z)</td>
</tr>
<tr>
<td>DESCENDING</td>
<td>Sort data in descending order (Z–A)</td>
</tr>
</tbody>
</table>

CrossFilter Class

Contains methods and constructors used to work with information about a cross filter.

Namespace

`Reports`

IN THIS SECTION:
- CrossFilter Constructors
- CrossFilter Methods

CrossFilter Constructors

The following are constructors for `CrossFilter`.

IN THIS SECTION:
- `CrossFilter(criteria, includesObject, primaryEntityField, relatedEntity, relatedEntityJoinField)`
  Creates an instance of the `Reports.CrossFilter` class using the specified parameters.
- `CrossFilter()`
  Creates an instance of the `Reports.CrossFilter` class. You can then set values by using the class's set methods.

**CrossFilter(criteria, includesObject, primaryEntityField, relatedEntity, relatedEntityJoinField)**

Creates an instance of the `Reports.CrossFilter` class using the specified parameters.

**Signature**

```java
public CrossFilter(List<Reports.ReportFilter> criteria, Boolean includesObject, String primaryEntityField, String relatedEntity, String relatedEntityJoinField)
```

**Parameters**

- `criteria`
  Type: `List<Reports.ReportFilter>`
Information about how to filter the `relatedEntity`. Relates the primary entity with a subset of the `relatedEntity`.

`includesObject`
Type: `Boolean`
Specifies whether objects returned have a relationship with the `relatedEntity` (`true`) or not (`false`).

`primaryEntityField`
Type: `String`
The name of the object on which the cross filter is evaluated.

`relatedEntity`
Type: `String`
The name of the object that the `primaryEntityField` is evaluated against—the right-hand side of the cross filter.

`relatedEntityJoinField`
Type: `String`
The name of the field used to join the `primaryEntityField` and `relatedEntity`.

**CrossFilter()**
Creates an instance of the `Reports.CrossFilter` class. You can then set values by using the class’s set methods.

**Signature**
```java
public CrossFilter()
```

**CrossFilter Methods**
The following are methods for `CrossFilter`.

**IN THIS SECTION:**
- `getCriteria()`
  Returns information about how to filter the `relatedEntity`. Describes the subset of the `relatedEntity` which the primary entity is evaluated against.
- `getIncludesObject()`
  Returns `true` if primary object has a relationship with the `relatedEntity`, otherwise returns `false`.
- `getPrimaryEntityField()`
  Returns the name of the object on which the cross filter is evaluated.
- `getRelatedEntity()`
  Returns name of the object that the `primaryEntityField` is evaluated against—the right-hand side of the cross filter.
- `getRelatedEntityJoinField()`
  Returns the name of the field used to join the `primaryEntityField` and `relatedEntity`.
- `setCriteria(criteria)`
  Specifies how to filter the `relatedEntity`. Relates the primary entity with a subset of the `relatedEntity`.
- `setIncludesObject(includesObject)`
  Specifies whether objects returned have a relationship with the `relatedEntity` (`true`) or not (`false`).
setPrimaryEntityField(primaryEntityField)
Specifies the name of the object on which the cross filter is evaluated.

setRelatedEntity(relatedEntity)
Specifies the name of the object that the primaryEntityField is evaluated against—the right-hand side of the cross filter.

setRelatedEntityJoinField(relatedEntityJoinField)
Specifies the name of the field used to join the primaryEntityField and relatedEntity.

toString()
Returns a string.

getCriteria() 
Returns information about how to filter the relatedEntity. Describes the subset of the relatedEntity which the primary entity is evaluated against.

Signature
public List<Reports.ReportFilter> getCriteria()

Return Value
Type: List<Reports.ReportFilter>

getIncludesObject() 
Returns true if primary object has a relationship with the relatedEntity, otherwise returns false.

Signature
public Boolean getIncludesObject()

Return Value
Type: Boolean

getPrimaryEntityField() 
Returns the name of the object on which the cross filter is evaluated.

Signature
public String getPrimaryEntityField()

Return Value
Type: String

getRelatedEntity() 
Returns name of the object that the primaryEntityField is evaluated against—the right-hand side of the cross filter.
Signature

public String getRelatedEntity()

Return Value

Type: String

getRelatedEntityJoinField()

Returns the name of the field used to join the primaryEntityField and relatedEntity.

Signature

public String getRelatedEntityJoinField()

Return Value

Type: String

setCriteria(criteria)

Specifies how to filter the relatedEntity. Relates the primary entity with a subset of the relatedEntity.

Signature

public void setCriteria(List<Reports.ReportFilter> criteria)

Parameters

criteria
Type: List<Reports.ReportFilter>

Return Value

Type: void

setIncludesObject(includesObject)

Specifies whether objects returned have a relationship with the relatedEntity (true) or not (false).

Signature

public void setIncludesObject(Boolean includesObject)

Parameters

includesObject
Type: Boolean
Return Value
Type: void

`setPrimaryEntityField(String primaryEntityField)`
Specifies the name of the object on which the cross filter is evaluated.

Signature
`public void setPrimaryEntityField(String primaryEntityField)`

Parameters
`primaryEntityField`
Type: `String`

Return Value
Type: void

`setRelatedEntity(String relatedEntity)`
Specifies the name of the object that the `primaryEntityField` is evaluated against—the right-hand side of the cross filter.

Signature
`public void setRelatedEntity(String relatedEntity)`

Parameters
`relatedEntity`
Type: `String`

Return Value
Type: void

`setRelatedEntityJoinField(String relatedEntityJoinField)`
Specifies the name of the field used to join the `primaryEntityField` and `relatedEntity`.

Signature
`public void setRelatedEntityJoinField(String relatedEntityJoinField)`

Parameters
`relatedEntityJoinField`
Type: `String`
Return Value
Type: void

**toString()**
Returns a string.

**Signature**
public String toString()

Return Value
Type: String

**CsfGroupType Enum**
The group level at which the custom summary format aggregate is displayed in a report.

**Enum Values**
The following are the values of the Reports.CsfGroupType enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALL</td>
<td>The aggregate is displayed at the end of every summary row.</td>
</tr>
<tr>
<td>CUSTOM</td>
<td>The aggregate is displayed at specified grouping levels.</td>
</tr>
<tr>
<td>GRAND_TOTAL</td>
<td>The aggregate is displayed only at the grand total level.</td>
</tr>
</tbody>
</table>

**DateGranularity Enum**
The Reports.DateGranularity enum describes the date interval that is used for grouping.

**Namespace**
Reports

**Usage**
The GroupingInfo.getDateGranularity method returns a Reports.DateGranularity enum value. The GroupingInfo.setDateGranularity method takes the enum value as an argument.

**Enum Values**
The following are the values of the Reports.DateGranularity enum.
<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DAY</td>
<td>The day of the week (Monday–Sunday)</td>
</tr>
<tr>
<td>DAY_IN_MONTH</td>
<td>The day of the month (1–31)</td>
</tr>
<tr>
<td>FISCAL_PERIOD</td>
<td>The fiscal period</td>
</tr>
<tr>
<td>FISCAL_QUARTER</td>
<td>The fiscal quarter</td>
</tr>
<tr>
<td>FISCAL_WEEK</td>
<td>The fiscal week</td>
</tr>
<tr>
<td>FISCAL_YEAR</td>
<td>The fiscal year</td>
</tr>
<tr>
<td>MONTH</td>
<td>The month (January–December)</td>
</tr>
<tr>
<td>MONTH_IN_YEAR</td>
<td>The month number (1–12)</td>
</tr>
<tr>
<td>NONE</td>
<td>No date grouping</td>
</tr>
<tr>
<td>QUARTER</td>
<td>The quarter number (1–4)</td>
</tr>
<tr>
<td>WEEK</td>
<td>The week number (1–52)</td>
</tr>
<tr>
<td>YEAR</td>
<td>The year number (###)</td>
</tr>
</tbody>
</table>

**DetailColumn Class**

Contains methods for describing fields that contain detailed data. Detailed data fields are also listed in the report metadata.

**Namespace**

*Reports*

**DetailColumn Instance Methods**

The following are instance methods for `DetailColumn`. All are instance methods.

**IN THIS SECTION:**

- `getName()`
  - Returns the unique API name of the detail column field.

- `getLabel()`
  - Returns the localized display name of a standard field, the ID of a custom field, or the API name of a bucket field that has detailed data.

- `getDataType()`
  - Returns the data type of a detail column field.

**getName()**

Returns the unique API name of the detail column field.
**Dimension Class**
Contains information for each row or column grouping.

**Namespace**
Reports

**Dimension Methods**
The following are methods for Dimension. All are instance methods.

IN THIS SECTION:
  - `getGroupings()`
    Returns information for each row or column grouping as a list.
getGroupings()
Returns information for each row or column grouping as a list.

Syntax
public List<Reports.GroupingValue> getGroupings()

Return Value
Type: List<Reports.GroupingValue>

EvaluatedCondition Class
Contains the individual components of an evaluated condition for a report notification, such as the aggregate name and label, the operator, and the value that the aggregate is compared to.

Namespace
Reports

IN THIS SECTION:
EvaluatedCondition Constructors
EvaluatedCondition Methods

EvaluatedCondition Constructors
The following are constructors for EvaluatedCondition.

IN THIS SECTION:
EvaluatedCondition(aggregateName, aggregateLabel, compareToValue, aggregateValue, displayCompareTo, displayValue, operator)
Creates a new instance of the Reports.EvaluatedConditions class using the specified parameters.

EvaluatedCondition(aggregateName, aggregateLabel, compareToValue, aggregateValue, displayCompareTo, displayValue, operator)
Creates a new instance of the Reports.EvaluatedConditions class using the specified parameters.

Signature
public EvaluatedCondition(String aggregateName, String aggregateLabel, Double compareToValue, Double aggregateValue, String displayCompareTo, String displayValue, Reports.EvaluatedConditionOperator operator)
Parameters

aggregateName
Type: String
The unique API name of the aggregate.

aggregateLabel
Type: String
The localized display name of the aggregate.

compareToValue
Type: Double
The value that the aggregate is compared to in the condition.

aggregateValue
Type: Double
The actual value of the aggregate when the report is run.

displayCompareTo
Type: String
The value that the aggregate is compared to in the condition, formatted for display. For example, a display value for a currency is $20.00 or USD20.00 instead of 20.00.

displayValue
Type: String
The value of the aggregate when the report is run, formatted for display. For example, a display value for a currency is $20.00 or USD20.00 instead of 20.00.

operator
Type: Reports.EvaluatedConditionOperator
The operator used in the condition.

EvaluatedCondition Methods

The following are methods for EvaluatedCondition.

IN THIS SECTION:

getAggregateLabel() Returns the localized display name of the aggregate.

getAggregateName() Returns the unique API name of the aggregate.

getCompareTo() Returns the value that the aggregate is compared to in the condition.

getDisplayCompareTo() Returns the value that the aggregate is compared to in the condition, formatted for display. For example, a display value for a currency is $20.00 or USD20.00 instead of 20.00.

getAddressValue() Returns the value of the aggregate when the report is run, formatted for display. For example, a display value for a currency is $20.00 or USD20.00 instead of 20.00.
getOperator()
Returns the operator used in the condition.

getValue()
Returns the actual value of the aggregate when the report is run.

getagregateLabel()  
Returns the localized display name of the aggregate.

Signature
public String getAggregateLabel()

Return Value
Type: String

getagregateName()  
Returns the unique API name of the aggregate.

Signature
public String getAggregateName()

Return Value
Type: String

getCompareTo()  
Returns the value that the aggregate is compared to in the condition.

Signature
public Double getCompareTo()

Return Value
Type: Double

getDisplayCompareTo()  
Returns the value that the aggregate is compared to in the condition, formatted for display. For example, a display value for a currency is $20.00 or USD20.00 instead of 20.00.

Signature
public String getDisplayCompareTo()
Return Value
Type: String

gDisplayValue()
Returns the value of the aggregate when the report is run, formatted for display. For example, a display value for a currency is $20.00 or USD20.00 instead of 20.00.

Signature
public String getDisplayValue()

Return Value
Type: String

gOperator()
Returns the operator used in the condition.

Signature
public Reports.EvaluatedConditionOperator getOperator()

Return Value
Type: Reports.EvaluatedConditionOperator

gValue()
Returns the actual value of the aggregate when the report is run.

Signature
public Double getValue()

Return Value
Type: Double

EvaluatedConditionOperator Enum
The Reports.EvaluatedConditionOperator enum describes the type of operator used to compare an aggregate to a value. It is returned by the getOperator method.

Namespace
Reports
**Enum Values**

The following are the values of the `Reports.EvaluatedConditionOperator` enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EQUAL</td>
<td>Equality operator.</td>
</tr>
<tr>
<td>GREATER_THAN</td>
<td>Greater than operator.</td>
</tr>
<tr>
<td>GREATER_THAN_EQUAL</td>
<td>Greater than or equal to operator.</td>
</tr>
<tr>
<td>LESS_THAN</td>
<td>Less than operator.</td>
</tr>
<tr>
<td>LESS_THAN_EQUAL</td>
<td>Less than or equal to operator.</td>
</tr>
<tr>
<td>NOT_EQUAL</td>
<td>Inequality operator.</td>
</tr>
</tbody>
</table>

**FilterOperator Class**

Contains information about a filter operator, such as display name and API name.

**Namespace**

`Reports`

**FilterOperator Methods**

The following are methods for `FilterOperator`. All are instance methods.

**IN THIS SECTION:**

- `getLabel()`
  Returns the localized display name of the filter operator. Possible values for this name are restricted based on the data type of the column being filtered.

- `getName()`
  Returns the unique API name of the filter operator. Possible values for this name are restricted based on the data type of the column being filtered. For example `multipicklist` fields can use the following filter operators: “equals,” “not equal to,” “includes,” and “excludes.” Bucket fields are considered to be of the `String` type.

**getLabel()**

Returns the localized display name of the filter operator. Possible values for this name are restricted based on the data type of the column being filtered.

**Syntax**

```java
public String getLabel()
```
Return Value
Type: String

getName()
Returns the unique API name of the filter operator. Possible values for this name are restricted based on the data type of the column being filtered. For example multipicklist fields can use the following filter operators: "equals," "not equal to," "includes," and "excludes." Bucket fields are considered to be of the String type.

Syntax
public String getName()
Return Value
Type: String

**getName()**
Returns the unique API name of the filter value. Possible values for this name are restricted based on the data type of the column being filtered.

**Syntax**

```java
public String getName()
```

Return Value
Type: String

**FormulaType Enum**
The format of the numbers in a custom summary formula.

**Enum Values**
The following are the values of the `Reports.FormulaType` enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CURRENCY</td>
<td>Formatted as currency. For example, $100.00.</td>
</tr>
<tr>
<td>NUMBER</td>
<td>Formatted as numbers. For example, 100.</td>
</tr>
<tr>
<td>PERCENT</td>
<td>Formatted as percentages. For example, 100%.</td>
</tr>
</tbody>
</table>

**GroupingColumn Class**
Contains methods for describing fields that are used for column grouping.

**Namespace**
`Reports`

The `GroupingColumn` class provides basic information about column grouping fields. The `GroupingInfo` class includes additional methods for describing and updating grouping fields.

**GroupingColumn Methods**
The following are methods for `GroupingColumn`. All are instance methods.
IN THIS SECTION:

- **getName()**
  Returns the unique API name of the field or bucket field that is used for column grouping.

- **getLabel()**
  Returns the localized display name of the field that is used for column grouping.

- **getDataType()**
  Returns the data type of the field that is used for column grouping.

- **getGroupingLevel()**
  Returns the level of grouping for the column.

**getName()**

Returns the unique API name of the field or bucket field that is used for column grouping.

**Syntax**

```java
public String getName()
```

**Return Value**

Type: String

**getLabel()**

Returns the localized display name of the field that is used for column grouping.

**Syntax**

```java
public String getLabel()
```

**Return Value**

Type: String

**getDataType()**

Returns the data type of the field that is used for column grouping.

**Syntax**

```java
public Reports.ColumnDataType getDataType()
```

**Return Value**

Type: Reports.ColumnDataType

**getGroupingLevel()**

Returns the level of grouping for the column.
Syntax

public Integer getGroupingLevel()

Return Value

Type: Integer

Usage

• In a summary report, 0, 1, or 2 indicates grouping at the first, second, or third row level.
• In a matrix report, 0 or 1 indicates grouping at the first or second row or column level.

GroupingInfo Class

Contains methods for describing fields that are used for grouping.

Namespace

Reports

GroupingInfo Methods

The following are methods for GroupingInfo. All are instance methods.

IN THIS SECTION:

getName()
Returns the unique API name of the field or bucket field that is used for row or column grouping.

getSortOrder()
Returns the order that is used to sort data in a row or column grouping (ASCENDING or DESCENDING).

dateGranularity()
Returns the date interval that is used for row or column grouping.

getSortAggregate()
Returns the summary field that is used to sort data within a grouping in a summary report. The value is null when data within a grouping is not sorted by a summary field.

getName()()
Returns the unique API name of the field or bucket field that is used for row or column grouping.

Syntax

public String getName()
**getSortOrder()**
Returns the order that is used to sort data in a row or column grouping (ASCENDING or DESCENDING).

**Syntax**
```java
public Reports.ColumnSortOrder getSortOrder()
```

**Return Value**
Type: Reports.ColumnSortOrder

**getDateGranularity()**
Returns the date interval that is used for row or column grouping.

**Syntax**
```java
public Reports.DateGranularity getDateGranularity()
```

**Return Value**
Type: Reports.DateGranularity

**getSortAggregate()**
Returns the summary field that is used to sort data within a grouping in a summary report. The value is null when data within a grouping is not sorted by a summary field.

**Syntax**
```java
public String getSortAggregate()
```

**Return Value**
Type: String

**GroupingValue Class**
Contains grouping values for a row or column, including the key, label, and value.

**Namespace**
Reports

**GroupingValue Methods**
The following are methods for GroupingValue. All are instance methods.
IN THIS SECTION:

**getGroupings()**
Returns a list of second- or third-level row or column groupings. If there are none, the value is an empty array.

**getKey()**
Returns the unique identifier for a row or column grouping. The identifier is used by the fact map to specify data values within each grouping.

**getLabel()**
Returns the localized display name of a row or column grouping. For date and time fields, the label is the localized date or time.

**getValue()**
Returns the value of the field that is used as a row or column grouping.

---

**getGroupings()**
Returns a list of second- or third-level row or column groupings. If there are none, the value is an empty array.

**Syntax**

```java
public LIST<Reports.GroupingValue> getGroupings()
```

**Return Value**

Type: `List<Reports.GroupingValue>`

---

**getKey()**
Returns the unique identifier for a row or column grouping. The identifier is used by the fact map to specify data values within each grouping.

**Syntax**

```java
public String getKey()
```

**Return Value**

Type: `String`

---

**getLabel()**
Returns the localized display name of a row or column grouping. For date and time fields, the label is the localized date or time.

**Syntax**

```java
public String getLabel()
```

**Return Value**

Type: `String`
**getValue()**

Returns the value of the field that is used as a row or column grouping.

**Syntax**

```java
public Object getValue()
```

**Return Value**

Type: Object

**Usage**

The value depends on the field’s data type.

- **Currency fields:**
  - `amount`: Of type currency. A data cell’s value.
  - `currency`: Of type picklist. The ISO 4217 currency code, if available; for example, USD for US dollars or CNY for Chinese yuan. (If the grouping is on the converted currency, this value is the currency code for the report and not for the record.)

- **Picklist fields:** API name. For example, a custom picklist field—Type of Business with values 1, 2, and 3 for Consulting, Services, and Add-On Business respectively—has 1, 2, or 3 as the grouping value.

- **ID fields:** API name.

- **Record type fields:** API name.

- **Date and time fields:** Date or time in ISO-8601 format.

- **Lookup fields:** Unique API name. For example, for the Opportunity Owner lookup field, the ID of each opportunity owner’s Chatter profile page can be a grouping value.

---

**NotificationAction Interface**

Implement this interface to trigger a custom Apex class when the conditions for a report notification are met.

**Namespace**

Reports

**Usage**

Report notifications for reports that users have subscribed to can trigger a custom Apex class, which must implement the Reports.NotificationAction interface. The execute method in this interface receives a NotificationActionContext object as a parameter, which contains information about the report instance and the conditions that must be met for a notification to be triggered.

IN THIS SECTION:

- NotificationAction Methods
- NotificationAction Example Implementation
NotificationAction Methods

The following are methods for NotificationAction.

IN THIS SECTION:

execute(context)

Executes the custom Apex action specified in the context parameter of the context object, NotificationActionContext. The object contains information about the report instance and the conditions that must be met for a notification to be triggered. The method executes whenever the specified conditions are met.

**execute (context)**

Executes the custom Apex action specified in the context parameter of the context object, NotificationActionContext. The object contains information about the report instance and the conditions that must be met for a notification to be triggered. The method executes whenever the specified conditions are met.

**Signature**

```java
public void execute(Reports.NotificationActionContext context)
```

**Parameters**

`context`

Type: Reports.NotificationActionContext

**Return Value**

Type: Void

NotificationAction Example Implementation

This is an example implementation of the Reports.NotificationAction interface.

```java
public class AlertOwners implements Reports.NotificationAction {

    public void execute(Reports.NotificationActionContext context) {
        Reports.ReportResults results = context.getReportInstance().getReportResults();
        for (Reports.GroupingValue g: results.getGroupingsDown().getGroupings()) {
            FeedItem t = new FeedItem();
            t.ParentId = (Id)g.getValue();
            t.Body = 'This record needs attention. Please view the report.';
            t.Title = 'Needs Attention: ' + results.getReportMetadata().getName();
            t.LinkUrl = '/' + results.getReportMetadata().getId();
            insert t;
        }
    }
}
```
NotificationActionContext Class

Contains information about the report instance and condition threshold for a report notification.

Namespace

Reports

IN THIS SECTION:
- NotificationActionContext Constructors
- NotificationActionContext Methods

NotificationActionContext Constructors

The following are constructors for NotificationActionContext.

IN THIS SECTION:
- NotificationActionContext(reportInstance, thresholdInformation)
  Creates a new instance of the Reports.NotificationActionContext class using the specified parameters.

**NotificationActionContext**(*reportInstance*, *thresholdInformation*)

Creates a new instance of the Reports.NotificationActionContext class using the specified parameters.

Signature

```java
public NotificationActionContext(Reports.ReportInstance reportInstance,
                                 Reports.ThresholdInformation thresholdInformation)
```

Parameters

- **reportInstance**
  - Type: Reports.ReportInstance
  - An instance of a report.
- **thresholdInformation**
  - Type: Reports.ThresholdInformation
  - The evaluated conditions for the notification.

NotificationActionContext Methods

The following are methods for NotificationActionContext.

IN THIS SECTION:
- **getReportInstance()**
  - Returns the report instance associated with the notification.
getThresholdInformation()
Returns the threshold information associated with the notification.

getReportInstance()
Returns the report instance associated with the notification.

Signature

public Reports.ReportInstance getReportInstance()

Return Value
Type: Reports.ReportInstance

getThresholdInformation()
Returns the threshold information associated with the notification.

Signature

public Reports.ThresholdInformation getThresholdInformation()

Return Value
Type: Reports.ThresholdInformation

ReportCsf Class
Contains methods and constructors for working with information about a custom summary formula (CSF).

Namespace
Reports

IN THIS SECTION:
ReportCsf Constructors
ReportCsf Methods

ReportCsf Constructors
The following are constructors for ReportCsf.

IN THIS SECTION:
ReportCsf(label, description, formulaType, decimalPlaces, downGroup, downGroupType, acrossGroup, acrossGroupType, formula)
Creates an instance of the Reports.ReportCsf class using the specified parameters.
ReportCsf()
Creates an instance of the Reports.ReportCsf class. You can then set values by using the class's set methods.

ReportCsf(label, description, formulaType, decimalPlaces, downGroup, downGroupType, acrossGroup, acrossGroupType, formula)
Creates an instance of the Reports.ReportCsf class using the specified parameters.

Signature
public ReportCsf(String label, String description, Reports.FormulaType formulaType, Integer decimalPlaces, String downGroup, Reports.CsfGroupType downGroupType, String acrossGroup, Reports.CsfGroupType acrossGroupType, String formula)

Parameters

label
Type: String
The user-facing name of the custom summary formula.

description
Type: String
The user-facing description of the custom summary formula.

formulaType
Type: Reports.FormulaType
The format of the numbers in the custom summary formula.

decimalPlaces
Type: Integer
The number of decimal places to include in numbers.

downGroup
Type: String
The name of a row grouping when the downGroupType is CUSTOM; null otherwise.

downGroupType
Type: Reports.CsfGroupType
Where to display the aggregate of the custom summary formula.

acrossGroup
Type: String
The name of a column grouping when the accrossGroupType is CUSTOM; null otherwise.

acrossGroupType
Type: Reports.CsfGroupType
Where to display the aggregate of the custom summary formula.

formula
Type: String
The operations performed on values in the custom summary formula.
**ReportCsf()**

Creates an instance of the `Reports.ReportCsf` class. You can then set values by using the class’s `set` methods.

**Signature**

```java
public ReportCsf()
```

**ReportCsf Methods**

The following are methods for `ReportCsf`.

**IN THIS SECTION:**

- `getAcrossGroup()`: Returns the name of a column grouping when the `acrossGroupType` is `CUSTOM`. Otherwise, returns `null`.  
- `getAcrossGroupType()`: Returns where to display the aggregate.  
- `getDecimalPlaces()`: Returns the number of decimal places that numbers in the custom summary formula have.  
- `getDescription()`: Returns the user-facing description of a custom summary formula.  
- `getDownGroup()`: Returns the name of a row grouping when the `downGroupType` is `CUSTOM`. Otherwise, returns `null`.  
- `getDownGroupType()`: Returns where to display the aggregate of the custom summary formula.  
- `getFormula()`: Returns the operations performed on values in the custom summary formula.  
- `getFormulaType()`: Returns the formula type.  
- `getLabel()`: Returns the user-facing name of the custom summary formula.  
- `setAcrossGroup(acrossGroup)`: Specifies the column for the across grouping.  
- `setAcrossGroupType(value)`: Sets where to display the aggregate.  
- `setAcrossGroupType(acrossGroupType)`: Sets where to display the aggregate.  
- `setDecimalPlaces(decimalPlaces)`: Sets the number of decimal places in numbers.  
- `setDescription(description)`: Sets the user-facing description of the custom summary formula.  
- `setDownGroup(downGroup)`: Sets the name of a row grouping when the `downGroupType` is `CUSTOM`.  

2574
setDownGroupType(value)
Sets where to display the aggregate.

setDownGroupType(downGroupType)
Sets where to display the aggregate.

setFormula(formula)
Sets the operations to perform on values in the custom summary formula.

setFormulaType(value)
Sets the format of the numbers in the custom summary formula.

setFormulaType(formulaType)
Sets the format of numbers used in the custom summary formula.

setLabel(label)
Sets the user-facing name of the custom summary formula.

toString()
Returns a string.

getAcrossGroup()
Returns the name of a column grouping when the acrossGroupType is CUSTOM. Otherwise, returns null.

Signature
public String getAcrossGroup()

Return Value
Type: String

getAcrossGroupType()
Returns where to display the aggregate.

Signature
public Reports.CsfGroupType getAcrossGroupType()

Return Value
Type: Reports.CsfGroupType

getDecimalPlaces()
Returns the number of decimal places that numbers in the custom summary formula have.

Signature
public Integer getDecimalPlaces()
Return Value
Type: Integer

description()
Returns the user-facing description of a custom summary formula.

Signature
public String getDescription()

Return Value
Type: String

downGroup()
Returns the name of a row grouping when the downGroupType is CUSTOM. Otherwise, returns null.

Signature
public String getDownGroup()

Return Value
Type: String

downGroupType()
Returns where to display the aggregate of the custom summary formula.

Signature
public Reports.CsfGroupType getDownGroupType()

Return Value
Type: Reports.CsfGroupType

formula()
Returns the operations performed on values in the custom summary formula.

Signature
public String getFormula()

Return Value
Type: String
getFormulaType ()
Returns the formula type.

Signature
public Reports.FormulaType getFormulaType()  

Return Value
Type: Reports.FormulaType

g.getLabel ()
Returns the user-facing name of the custom summary formula.

Signature
public String getLabel()  

Return Value
Type: String

setAcrossGroup (acrossGroup)
Specifies the column for the across grouping.

Signature
public void setAcrossGroup(String acrossGroup)  

Parameters
acrossGroup
Type: String

Return Value
Type: void

setAcrossGroupType (value)
Sets where to display the aggregate.

Signature
public void setAcrossGroupType(String value)
Parameters

\texttt{value}

Type: String

For possible values, see \texttt{Reports.CsfGroupType}.

Return Value

Type: void

\texttt{setAcrossGroupType(\texttt{acrossGroupType})}

Sets where to display the aggregate.

Signature

\texttt{public void setAcrossGroupType(Reports.CsfGroupType \texttt{acrossGroupType})}

Parameters

\texttt{acrossGroupType}

Type: \texttt{Reports.CsfGroupType}

Return Value

Type: void

\texttt{setDecimalPlaces(\texttt{decimalPlaces})}

Sets the number of decimal places in numbers.

Signature

\texttt{public void setDecimalPlaces(Integer \texttt{decimalPlaces})}

Parameters

\texttt{decimalPlaces}

Type: \texttt{Integer}

Return Value

Type: void

\texttt{setDescription(\texttt{description})}

Sets the user-facing description of the custom summary formula.

Signature

\texttt{public void setDescription(String \texttt{description})}
Parameters
description
Type: String

Return Value
Type: void

setDownGroup (downGroup)
Sets the name of a row grouping when the downGroupType is CUSTOM.

Signature
public void setDownGroup(String downGroup)

Parameters
downGroup
Type: String

Return Value
Type: void

setDownGroupType (value)
Sets where to display the aggregate.

Signature
public void setDownGroupType(String value)

Parameters
value
Type: String
For valid values, see Reports.CsfGroupType.

Return Value
Type: void

setDownGroupType (downGroupType)
Sets where to display the aggregate.

Signature
public void setDownGroupType(Reports.CsfGroupType downGroupType)
Parameters

downGroupType
   Type: Reports.CsfGroupType

Return Value
Type: void

setFormula(formula)
Sets the operations to perform on values in the custom summary formula.

Signature
public void setFormula(String formula)

Parameters

formula
   Type: String

Return Value
Type: void

setFormulaType(value)
Sets the format of the numbers in the custom summary formula.

Signature
public void setFormulaType(String value)

Parameters

value
   Type: String
   For valid values, see Reports.FormulaType.

Return Value
Type: void

setFormulaType(formulaType)
Sets the format of numbers used in the custom summary formula.

Signature
public void setFormulaType(Reports.FormulaType formulaType)
Parameters

\textit{formulaType}

Type: \texttt{Reports.FormulaType}

Return Value

Type: void

\texttt{setLabel(label)}

Sets the user-facing name of the custom summary formula.

Signature

\texttt{public void setLabel(String label)}

Parameters

\textit{label}

Type: \texttt{String}

Return Value

Type: void

\texttt{toString()}

Returns a string.

Signature

\texttt{public String toString()}

Return Value

Type: \texttt{String}

\textbf{ReportCurrency Class}

Contains information about a currency value, including the amount and currency code.

\textbf{Namespace}

\texttt{Reports}

\textbf{ReportCurrency Methods}

The following are methods for \texttt{ReportCurrency}. All are instance methods.
getAmount()  
Returns the amount of the currency value.

getCurrencyCode()  
Returns the report currency code, such as USD, EUR, or GBP, for an organization that has multicurrency enabled. The value is null if the organization does not have multicurrency enabled.

getAmount()  
Returns the amount of the currency value.

Syntax  
public Decimal getAmount()  

Return Value  
Type: Decimal  

getCurrencyCode()  
Returns the report currency code, such as USD, EUR, or GBP, for an organization that has multicurrency enabled. The value is null if the organization does not have multicurrency enabled.

Syntax  
public String getCurrencyCode()  

Return Value  
Type: String  

ReportDataCell Class  
Contains the data for a cell in the report, including the display label and value.

Namespace  
Reports  

ReportDataCell Methods  
The following are methods for ReportDataCell. All are instance methods.

IN THIS SECTION:  
getLabel()  
Returns the localized display name of the value of a specified cell in the report.
getValue()
Returns the value of a specified cell of a detail row of a report.

getLabel()
Returns the localized display name of the value of a specified cell in the report.

Syntax
public String getLabel()

Return Value
Type: String

g getValue ()
Returns the value of a specified cell of a detail row of a report.

Syntax
public Object getValue ()

Return Value
Type: Object

ReportDescribeResult Class
Contains report, report type, and extended metadata for a tabular, summary, or matrix report.

Namespace
Reports

ReportDescribeResult Methods
The following are methods for ReportDescribeResult. All are instance methods.

IN THIS SECTION:
  getReportExtendedMetadata()
  Returns additional information about grouping and summaries.
  getReportMetadata()
  Returns unique identifiers for groupings and summaries.
  getReportTypeMetadata()
  Returns the fields in each section of a report type, plus filtering information for those fields.
getReportExtendedMetadata()
Returns additional information about grouping and summaries.

Syntax
public Reports.ReportExtendedMetadata getReportExtendedMetadata()

Return Value
Type: Reports.ReportExtendedMetadata

getReportMetadata()
Returns unique identifiers for groupings and summaries.

Syntax
public Reports.ReportMetadata getReportMetadata()

Return Value
Type: Reports.ReportMetadata

getReportTypeMetadata()
Returns the fields in each section of a report type, plus filtering information for those fields.

Syntax
public Reports.ReportTypeMetadata getReportTypeMetadata()

Return Value
Type: Reports.ReportTypeMetadata

ReportDetailRow Class
Contains data cells for a detail row of a report.

Namespace
Reports

ReportDetailRow Methods
The following are methods for ReportDetailRow. All are instance methods.
IN THIS SECTION:

**getDataCells()**
Returns a list of data cells for a detail row.

**getDataCells()**
Returns a list of data cells for a detail row.

Syntax

```java
public LIST<Reports.ReportDataCell> getDataCells()
```

Return Value

Type: List<Reports.ReportDataCell>

**ReportDivisionInfo Class**

Contains information about the divisions that can be used to filter a report.

Available only if your organization uses divisions to segment data and you have the “Affected by Divisions” permission. If you do not have the “Affected by Divisions” permission, your reports include records in all divisions.

**Namespace**

Reports

**Usage**

Use to filter records in the report based on a division, like West Coast and East Coast.

**ReportDivisionInfo Methods**

The following are methods for ReportDivisionInfo.

**getDefaultValue()**
Returns the default division for the report.

**Signature**

```java
public String getDefaultValue()
```

**Return Value**

Type: String

**getValues()**
Returns a list of all possible divisions for the report.
**Signature**

```java
public List<Reports.FilterValue> getValues()
```

**Return Value**

Type: `List<Reports.FilterValue>`

---

### ReportExtendedMetadata Class

Contains report extended metadata for a tabular, summary, or matrix report.

---

### Namespace

**Reports**

Report extended metadata provides additional, detailed metadata about summary and grouping fields, including data type and label information.

---

### ReportExtendedMetadata Methods

The following are methods for `ReportExtendedMetadata`. All are instance methods.

**IN THIS SECTION:**

- `getAggregateColumnInfo()`
  - Returns all report summaries such as `Record Count`, `Sum`, `Average`, `Max`, `Min`, and custom summary formulas. Contains values for each summary that is listed in the report metadata.

- `getDetailColumnInfo()`
  - Returns a map of two properties for each field that has detailed data identified by its unique API name. The detailed data fields are also listed in the report metadata.

- `getGroupingColumnInfo()`
  - Returns a map of each row or column grouping to its metadata. Contains values for each grouping that is identified in the `groupingsDown` and `groupingsAcross` lists.

**getAggregateColumnInfo()**

Returns all report summaries such as `Record Count`, `Sum`, `Average`, `Max`, `Min`, and custom summary formulas. Contains values for each summary that is listed in the report metadata.

**Syntax**

```java
public MAP<String,Reports.AggregateColumn> getAggregateColumnInfo()
```

**Return Value**

Type: `Map<String,Reports.AggregateColumn>`

---

2586
getDetailColumnInfo()

Returns a map of two properties for each field that has detailed data identified by its unique API name. The detailed data fields are also listed in the report metadata.

Syntax

public MAP<String,Reports.DetailColumn> getDetailColumnInfo()

Return Value

Type: Map<String,Reports.DetailColumn>

getGroupingColumnInfo()

Returns a map of each row or column grouping to its metadata. Contains values for each grouping that is identified in the groupingsDown and groupingsAcross lists.

Syntax

public MAP<String,Reports.GroupingColumn> getGroupingColumnInfo()

Return Value

Type: Map<String,Reports.GroupingColumn>

ReportFact Class

Contains the fact map for the report, which represents the report's data values.

Namespace

Reports

Usage

ReportFact is the parent class of ReportFactWithDetails and ReportFactWithSummaries. If includeDetails is true when the report is run, the fact map is a ReportFactWithDetails object. If includeDetails is false when the report is run, the fact map is a ReportFactWithSummaries object.

ReportFact Methods

The following are methods for ReportFact. All are instance methods.

IN THIS SECTION:

getaGgregates()

Returns summary-level data for a report, including the record count.
getKey()
Returns the unique identifier for a row or column grouping. This identifier can be used to index specific data values within each grouping.

def getKey()
Returns summary-level data for a report, including the record count.

def getAggregates()

Syntax
public LIST<Reports.SummaryValue> getAggregates()

Return Value
Type: List<Reports.SummaryValue>

getKey()
Returns the unique identifier for a row or column grouping. This identifier can be used to index specific data values within each grouping.

Syntax
public String getKey()

Return Value
Type: String

ReportFactWithDetails Class
Contains the detailed fact map for the report, which represents the report’s data values.

Namespace
Reports

Usage
The ReportFactWithDetails class extends the ReportFact class. A ReportFactWithDetails object is returned if includeDetails is set to true when the report is run. To access the detail values, you’ll need to cast the return value of the ReportResults.getFactMap method to a ReportFactWithDetails object.

ReportFactWithDetails Methods
The following are methods for ReportFactWithDetails. All are instance methods.
IN THIS SECTION:

getAggregates()
Returns summary-level data for a report, including the record count.

getKey()
Returns the unique identifier for a row or column grouping. This identifier can be used to index specific data values within each grouping.

getRows()
Returns a list of detailed report data in the order of the detail columns that are provided by the report metadata.

getAggregates()
Returns summary-level data for a report, including the record count.

Syntax
public LIST<Reports.SummaryValue> getAggregates()

Return Value
Type: List<Reports.SummaryValue>

getKey()
Returns the unique identifier for a row or column grouping. This identifier can be used to index specific data values within each grouping.

Syntax
public String getKey()

Return Value
Type: String

getRows()
Returns a list of detailed report data in the order of the detail columns that are provided by the report metadata.

Syntax
public LIST<Reports.ReportDetailRow> getRows()

Return Value
Type: List<Reports.ReportDetailRow>

ReportFactWithSummaries Class
Contains the fact map for the report, which represents the report’s data values, and includes summarized fields.
Namespace
Reports

Usage
The ReportFactWithSummaries class extends the ReportFact class. A ReportFactWithSummaries object is returned if includeDetails is set to false when the report is run.

ReportFactWithSummaries Methods
The following are methods for ReportFactWithSummaries. All are instance methods.

IN THIS SECTION:
getAggregates()
Returns summary-level data for a report, including the record count.
getKey()
Returns the unique identifier for a row or column grouping. This identifier can be used to index specific data values within each grouping.
toString()
Returns a string.

getAggregates()
Returns summary-level data for a report, including the record count.

Syntax
public LIST<Reports.SummaryValue> getAggregates()

Return Value
Type: List<Reports.SummaryValue>

getKey()

Syntax
public String getKey()

Return Value
Type: String
toString()
Returns a string.

Signature
public String toString()

Return Value
Type: String

ReportFilter Class
Contains information about a report filter, including column, operator, and value.

Namespace
Reports

IN THIS SECTION:
ReportFilter Constructors
ReportFilter Methods

ReportFilter Constructors
The following are constructors for ReportFilter.

IN THIS SECTION:
ReportFilter()
Creates a new instance of the Reports.ReportFilter class. You can then set values by using the “set” methods.
ReportFilter(column, operator, value)
Creates a new instance of the Reports.ReportFilter class by using the specified parameters.
ReportFilter(column, operator, value, filterType)
Creates a new instance of the Reports.ReportFilter class by using the specified parameters.

ReportFilter()
Creates a new instance of the Reports.ReportFilter class. You can then set values by using the “set” methods.

Signature
public ReportFilter()

ReportFilter(column, operator, value)
Creates a new instance of the Reports.ReportFilter class by using the specified parameters.
Signature

public ReportFilter(String column, String operator, String value)

Parameters

column
  Type: String
operator
  Type: String
value
  Type: String

ReportFilter(column, operator, value, filterType)

Creates a new instance of the Reports.ReportFilter class by using the specified parameters.

Syntax

public ReportFilterType(String column, String operator, String value, Reports.ReportFilterType filterType)

Parameters

column
  Type: String
operator
  Type: String
value
  Type: String
filterType
  Type: ReportFilterType Enum on page 2596

ReportFilter Methods

The following are methods for ReportFilter. All are instance methods.

IN THIS SECTION:

getColumn()
  Returns the unique API name for the field that’s being filtered.
getFilterType()
  Returns the type of report filter.
getOperator()
  Returns the unique API name for the condition that is used to filter a field, such as “greater than” or “not equal to.” Filter conditions depend on the data type of the field.
getValue()
Returns the value that the field is being filtered by. For example, the field `Age` can be filtered by a numeric value.

setColumn(column)
Sets the unique API name for the field that's being filtered.

setFilterType()
Sets the type of report filter.

setOperator(operator)
Sets the unique API name for the condition that is used to filter a field, such as "greater than" or "not equal to." Filter conditions depend on the data type of the field.

setValue(value)
Sets the value by which a field can be filtered. For example, the field `Age` can be filtered by a numeric value.

toString(column)
Returns a string representation of the filter.

getColumn()
Returns the unique API name for the field that's being filtered.

**Syntax**
```
public String getColumn()
```

**Return Value**
Type: `String`

getFilterType()
Returns the type of report filter.

**Syntax**
```
public String getFilterType()
```

**Return Value**
Type: `ReportFilterType Enum` on page 2596

getOperator()
Returns the unique API name for the condition that is used to filter a field, such as "greater than" or "not equal to." Filter conditions depend on the data type of the field.

**Syntax**
```
public String getOperator()
```
Return Value
Type: String

**getValue()**
Returns the value that the field is being filtered by. For example, the field *Age* can be filtered by a numeric value.

Syntax
```java
public String getValue()
```

---

**setColumn(column)**
Sets the unique API name for the field that’s being filtered.

Syntax
```java
public Void setColumn(String column)
```

Parameters
- **column**
  Type: String

Return Value
Type: Void

---

**setFilterType()**
Sets the type of report filter.

Syntax
```java
public Void setFilterType(String column)
```

Parameters
- **column**
  Type: String

Return Value
Type: Void
**setOperator**(operator)
Sets the unique API name for the condition that is used to filter a field, such as "greater than" or "not equal to." Filter conditions depend on the data type of the field.

**Syntax**
```
public Void setOperator(String operator)
```

**Parameters**
- **operator**
  Type: String

**Return Value**
Type: Void

**setValue**(value)
Sets the value by which a field can be filtered. For example, the field **Age** can be filtered by a numeric value.

**Syntax**
```
public Void setValue(String value)
```

**Parameters**
- **value**
  Type: String

**Return Value**
Type: Void

**toString**(column)
Returns a string representation of the filter.

**Signature**
```
public String toString()
```

**Return Value**
Type: String

**ReportFormat Enum**
Contains the possible report format types.
Namespace
Reports

Enum Values
The following are the values of the Reports.ReportFormat enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MATRIX</td>
<td>Matrix report format</td>
</tr>
<tr>
<td>SUMMARY</td>
<td>Summary report format</td>
</tr>
<tr>
<td>TABULAR</td>
<td>Tabular report format</td>
</tr>
</tbody>
</table>

ReportFilterType Enum
The types of values included in a report filter type.

Enum Values
The following are the values of the Reports.ReportFilterType enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>fieldToField</td>
<td>Field-to-field filter</td>
</tr>
<tr>
<td>fieldValue</td>
<td>Field-to-value filter</td>
</tr>
</tbody>
</table>

ReportInstance Class
Returns an instance of a report that was run asynchronously. Retrieves the results for that instance.

Namespace
Reports

ReportInstance Methods
The following are methods for ReportInstance. All are instance methods.

IN THIS SECTION:
- `getCompletionDate()`
  Returns the date and time when the instance of the report finished running. The completion date is available only if the report instance ran successfully or couldn’t be run because of an error. Date and time information is in ISO-8601 format.
- `getId()`
  Returns the unique ID for an instance of a report that was run asynchronously.
getOwnerId()  
Returns the ID of the user who created the report instance.

getReportId()  
Returns the unique ID of the report this instance is based on.

getReportResults()  
Retrieves results for an instance of an asynchronous report. When you request your report, you can specify whether to summarize data or include details.

getRequestDate()  
Returns the date and time when an instance of the report was run. Date and time information is in ISO-8601 format.

getStatus()  
Returns the status of a report.

getCompletionDate()  
Returns the date and time when the instance of the report finished running. The completion date is available only if the report instance ran successfully or couldn’t be run because of an error. Date and time information is in ISO-8601 format.

Syntax

```java
public Datetime getCompletionDate()
```

Return Value

Type: Datetime

ggetId()  
Returns the unique ID for an instance of a report that was run asynchronously.

Syntax

```java
public Id getId()
```

Return Value

Type: Id

ggetOwnerId()  
Returns the ID of the user who created the report instance.

Syntax

```java
public Id getOwnerId()
```

Return Value

Type: Id
**getReportId()**
Returns the unique ID of the report this instance is based on.

**Syntax**
```
public Id getReportId()
```

**Return Value**
Type: Id

**getReportResults()**
Retrieves results for an instance of an asynchronous report. When you request your report, you can specify whether to summarize data or include details.

**Syntax**
```
public Reports.ReportResults getReportResults()
```

**Return Value**
Type: Reports.ReportResults

**getRequestDate()**
Returns the date and time when an instance of the report was run. Date and time information is in ISO-8601 format.

**Syntax**
```
public Datetime getRequestDate()
```

**Return Value**
Type: Datetime

**getStatus()**
Returns the status of a report.

**Syntax**
```
public String getStatus()
```

**Return Value**
Type: String
Usage

- **New** if the report run was recently triggered through a request.
- **Success** if the report ran.
- **Running** if the report is being run.
- **Error** if the report run failed. The instance of a report run can return an error if, for example, your permission to access the report was removed after you requested the run.

ReportManager Class

Runs a report synchronously or asynchronously and with or without details.

Namespace

Reports

Usage

Gets instances of reports and describes the metadata of Reports.

ReportManager Methods

The following are methods for ReportManager. All methods are static.

IN THIS SECTION:

- `describeReport(reportId)`
  
  Retrieves report, report type, and extended metadata for a tabular, summary, or matrix report.

- `getDatatypeFilterOperatorMap()`
  
  Lists the field data types that you can use to filter the report.

- `getReportInstance(instanceld)`
  
  Retrieves results for an instance of a report that has been run asynchronously. The settings you use when you run your asynchronous report determine whether you can retrieve summary data or detailed data.

- `getReportInstances(reportId)`
  
  Returns a list of instances for a report that was run asynchronously. Each item in the list represents a separate instance of the report, with metadata for the time at which the report was run.

- `runAsyncReport(reportId, reportMetadata, includeDetails)`
  
  Runs a report asynchronously with the report ID. Includes details if `includeDetails` is set to `true`. Filters the report based on the report metadata in `reportMetadata`.

- `runAsyncReport(reportId, includeDetails)`
  
  Runs a report asynchronously with the report ID. Includes details if `includeDetails` is set to `true`.

- `runAsyncReport(reportId, reportMetadata)`
  
  Runs a report asynchronously with the report ID. Filters the results based on the report metadata in `reportMetadata`.

- `runAsyncReport(reportId)`
  
  Runs a report asynchronously with the report ID.
Run a report immediately with the report ID. Includes details if `includeDetails` is set to `true`. Filters the results based on the report metadata in `reportMetadata`.

```
runReport(reportId, reportMetadata, includeDetails)
```

Runs a report immediately with the report ID. Includes details if `includeDetails` is set to `true`.

```
runReport(reportId, includeDetails)
```

Runs a report immediately with the report ID. Filters the results based on the report metadata in `reportMetadata`.

```
runReport(reportId, reportMetadata)
```

Runs a report immediately with the report ID.

```
runReport(reportId)
```

`describeReport(reportId)`

Retrieves report, report type, and extended metadata for a tabular, summary, or matrix report.

**Syntax**

```
public static Reports.ReportDescribeResult describeReport(Id reportId)
```

**Parameters**

- `reportId`
  - Type: `Id`

**Return Value**

- Type: `Reports.ReportDescribeResult`

`getDatatypeFilterOperatorMap()`

Lists the field data types that you can use to filter the report.

**Syntax**

```
public static MAP<String, LIST<Reports.FilterOperator>> getDatatypeFilterOperatorMap()
```

**Return Value**

- Type: `Map<String, List<Reports.FilterOperator>>`

`getReportInstance(instanceId)`

Retrieves results for an instance of a report that has been run asynchronously. The settings you use when you run your asynchronous report determine whether you can retrieve summary data or detailed data.

**Syntax**

```
public static Reports.ReportInstance getReportInstance(Id instanceId)
```
Parameters

instanceId
Type: Id

Return Value
Type: Reports.ReportInstance

getReportInstances (reportId)
Returns a list of instances for a report that was run asynchronously. Each item in the list represents a separate instance of the report, with metadata for the time at which the report was run.

Syntax
public static LIST<Reports.ReportInstance> getReportInstances(Id reportId)

Parameters
reportId
Type: Id

Return Value
Type: List<Reports.ReportInstance>

runAsyncReport (reportId, reportMetadata, includeDetails)
Runs a report asynchronously with the report ID. Includes details if includeDetails is set to true. Filters the report based on the report metadata in reportMetadata.

Syntax
public static Reports.ReportInstance runAsyncReport(Id reportId, Reports.ReportMetadata reportMetadata, Boolean includeDetails)

Parameters
reportId
Type: Id
reportMetadata
Type: Reports.ReportMetadata
includeDetails
Type: Boolean

Return Value
Type: Reports.ReportInstance
runAsyncReport(reportId, includeDetails)
Runs a report asynchronously with the report ID. Includes details if includeDetails is set to true.

Syntax
public static Reports.ReportInstance runAsyncReport(Id reportId, Boolean includeDetails)

Parameters
reportId
Type: Id
includeDetails
Type: Boolean

Return Value
Type: Reports.ReportInstance

runAsyncReport(reportId, reportMetadata)
Runs a report asynchronously with the report ID. Filters the results based on the report metadata in reportMetadata.

Syntax
public static Reports.ReportInstance runAsyncReport(Id reportId, Reports.ReportMetadata reportMetadata)

Parameters
reportId
Type: Id
reportMetadata
Type: Reports.ReportMetadata

Return Value
Type: Reports.ReportInstance

runAsyncReport(reportId)
Runs a report asynchronously with the report ID.

Syntax
public static Reports.ReportInstance runAsyncReport(Id reportId)
Parameters

reportId
Type: Id

Return Value

Type: Reports.ReportInstance

runReport(reportId, reportMetadata, includeDetails)
Runs a report immediately with the report ID. Includes details if includeDetails is set to true. Filters the results based on the report metadata in reportMetadata.

Syntax

public static Reports.ReportResults runReport(Id reportId, Reports.ReportMetadata reportMetadata, Boolean includeDetails)

Parameters

reportId
Type: Id
reportMetadata
Type: Reports.ReportMetadata
includeDetails
Type: Boolean

Return Value

Type: Reports.ReportResults

runReport(reportId, includeDetails)
Runs a report immediately with the report ID. Includes details if includeDetails is set to true.

Syntax

public static Reports.ReportResults runReport(Id reportId, Boolean includeDetails)
runReport(reportId, reportMetadata)

Runs a report immediately with the report ID. Filters the results based on the report metadata in `rmData`.

**Syntax**

```java
public static Reports.ReportResults runReport(Id reportId, Reports.ReportMetadata reportMetadata)
```

**Parameters**

`reportId`
Type: `Id`

`reportMetadata`
Type: `Reports.ReportMetadata`

**Return Value**
Type: `Reports.ReportResults`

runReport(reportId)

Runs a report immediately with the report ID.

**Syntax**

```java
public static Reports.ReportResults runReport(Id reportId)
```

**Parameters**

`reportId`
Type: `Id`

**Return Value**
Type: `Reports.ReportResults`

**ReportMetadata Class**

Contains report metadata for a tabular, summary, or matrix report.

**Namespace**

`Reports`

**Usage**

Report metadata gives information about the report as a whole, such as the report type, format, summary fields, row or column groupings, and filters that are saved to the report. You can use the `ReportMetadata` class to retrieve report metadata and to set metadata that can be used to filter a report.
ReportMetadata Methods

The following are methods for ReportMetadata. All are instance methods.

IN THIS SECTION:

- getAggregates()
  Returns unique identifiers for summary or custom summary formula fields in the report.

- getBuckets()
  Returns a list of bucket fields in the report.

- getCrossFilters()
  Returns information about cross filters applied to a report.

- getCurrencyCode()
  Returns report currency, such as USD, EUR, or GBP, for an organization that has multicurrency enabled. The value is null if the organization does not have multicurrency enabled.

- getCustomSummaryFormula()
  Returns information about custom summary formulas in a report.

- getDescription()
  Returns the description of the report.

- getDetailColumns()
  Returns unique API names (column names) for the fields that contain detailed data. For example, the method might return the following values: "OPPORTUNITY_NAME, TYPE, LEAD_SOURCE, AMOUNT."

- getDeveloperName()
  Returns the report API name. For example, the method might return the following value: "Closed_Sales_This_Quarter."

- getDivision()
  Returns the division specified in the report.

- getGroupingsAcross()
  Returns column groupings in a report.

- getGroupingsDown()
  Returns row groupings for a report.

- getHasDetailRows()
  Indicates whether the report has detail rows.

- getHasRecordCount()
  Indicates whether the report shows the total number of records.

- getHistoricalSnapshotDates()
  Returns a list of historical snapshot dates.

- getId()
  Returns the unique report ID.

- getName()
  Returns the report name.

- getReportBooleanFilter()
  Returns logic to parse custom field filters. The value is null when filter logic is not specified.
getReportFilters()
Returns a list of each custom filter in the report along with the field name, filter operator, and filter value.

getReportFormat()
Returns the format of the report.

getReportType()
Returns the unique API name and display name for the report type.

getScope()
Returns the API name for the scope defined for the report. Scope values depend on the report type.

getShowGrandTotal()
Indicates whether the report shows the grand total.

getShowSubtotals()
Indicates whether the report shows subtotals, such as column or row totals.

getSortBy()
Returns the list of columns on which the report is sorted. Currently, you can sort on only one column.

getStandardDateFilter()
Returns information about the standard date filter for the report, such as the start date, end date, date range, and date field API name.

getStandardFilters()
Returns a list of standard filters for the report.

getTopRows()
Returns information about a row limit filter, including the number of rows returned and the sort order.

setAggregates(aggregates)
Sets unique identifiers for standard or custom summary formula fields in the report.

setBuckets(buckets)
Creates bucket fields in a report.

setCrossFilters(crossFilters)
Applies cross filters to a report.

setCurrencyCode(currencyCode)
Sets the currency, such as USD, EUR, or GBP, for report summary fields in an organization that has multicurrency enabled.

setCustomSummaryFormula(customSummaryFormula)
Adds a custom summary formula to a report.

setDescription(description)
Sets the description of the report.

setDetailColumns(detailColumns)
Sets the unique API names for the fields that contain detailed data—for example, OPPORTUNITY_NAME, TYPE, LEAD_SOURCE, or AMOUNT.

setDeveloperName(developerName)
Sets the report API name—for example, Closed_Sales_This_Quarter.

setDivision(division)
Sets the division of the report.
setGroupingsAcross(groupingInfo)
Sets column groupings in a report.

setGroupingsDown(groupingInfo)
Sets row groupings for a report.

setHasDetailRows(hasDetailRows)
Specifies whether the report has detail rows.

setHasRecordCount(hasRecordCount)
Specifies whether the report is configured to show the total number of records.

setHistoricalSnapshotDates(historicalSnapshot)
Sets a list of historical snapshot dates.

setId(id)
Sets the unique report ID.

setName(name)
Sets the report name.

setReportBooleanFilter(reportBooleanFilter)
Sets logic to parse custom field filters.

setReportFilters(reportFilters)
Sets a list of each custom filter in the report along with the field name, filter operator, and filter value.

setReportFormat(format)
Sets the format of the report.

setReportType(reportType)
Sets the unique API name and display name for the report type.

setScope(scopeName)
Sets the API name for the scope defined for the report. Scope values depend on the report type.

setShowGrandTotal(showGrandTotal)
Specifies whether the report shows the grand total.

setShowSubtotals(showSubtotals)
Specifies whether the report shows subtotals, such as column or row totals.

setSortBy(column)
Sets the list of columns on which the report is sorted. Currently, you can only sort on one column.

setStandardDateFilter(dateFilter)
Sets the standard date filter—which includes the start date, end date, date range, and date field API name—for the report.

setStandardFilters(filters)
Sets one or more standard filters on the report.

setTopRows(topRows)
Applies a row limit filter to a report.

getAggregates()
Returns unique identifiers for summary or custom summary formula fields in the report.
Syntax
public LIST<String> getAggregates()

Return Value
Type: List<String>

Usage
For example:
- a!Amount represents the average for the Amount column.
- s!Amount represents the sum of the Amount column.
- m!Amount represents the minimum value of the Amount column.
- x!Amount represents the maximum value of the Amount column.
- s! <customfieldID> represents the sum of a custom field column. For custom fields and custom report types, the identifier is a combination of the summary type and the field ID.

getBuckets()
Returns a list of bucket fields in the report.

Signature
public List<Reports.BucketField> getBuckets()

Return Value
Type: List<Reports.BucketField>

getcrossfilters()
Returns information about cross filters applied to a report.

Signature
public Reports.CrossFilter getCrossFilters()

Return Value
Type: List<Reports.CrossFilter>

getchannel()

getCurrencyCode()
Returns report currency, such as USD, EUR, or GBP, for an organization that has multicurrency enabled. The value is null if the organization does not have multicurrency enabled.

Syntax
public String getCurrencyCode()
getCustomSummaryFormula()
Returns information about custom summary formulas in a report.

Signature
public Map<String, Reports.ReportCsf> getCustomSummaryFormula()

Return Value
Type: Map<String, Reports.ReportCsf>

description()
Returns the description of the report.

Signature
public String getDescription()

Return Value
Type: String

detailColumns()
Returns unique API names (column names) for the fields that contain detailed data. For example, the method might return the following values: "OPPORTUNITY_NAME, TYPE, LEAD_SOURCE, AMOUNT."

Syntax
public List<String> getDetailColumns()

Return Value
Type: List<String>

developerName()
Returns the report API name. For example, the method might return the following value: "Closed_Sales_This_Quarter."

Syntax
public String getDeveloperName()
Return Value
Type: String

getDivision()
Returns the division specified in the report.

Note: Reports that use standard filters (such as My Cases or My Team’s Accounts) show records in all divisions. These reports can’t be further limited to a specific division.

Signature
public String getDivision()

Return Value
Type: String

getGroupingsAcross()
Returns column groupings in a report.

Syntax
public LIST<Reports.GroupingInfo> getGroupingsAcross()

Return Value
Type: List<Reports.GroupingInfo>

Usage
The identifier is:
• An empty array for reports in summary format, because summary reports don’t include column groupings
• BucketField_(ID) for bucket fields
• The ID of a custom field when the custom field is used for a column grouping

getGroupingsDown()
Returns row groupings for a report.

Syntax
public LIST<Reports.GroupingInfo> getGroupingsDown()

Return Value
Type: List<Reports.GroupingInfo>
Usage
The identifier is:

- BucketField_\(\text{ID}\) for bucket fields
- The ID of a custom field when the custom field is used for grouping

**getHasDetailRows()**
Indicates whether the report has detail rows.

**Signature**
```
public Boolean getHasDetailRows()
```

**Return Value**
Type: Boolean

**getHasRecordCount()**
Indicates whether the report shows the total number of records.

**Signature**
```
public Boolean getHasRecordCount()
```

**Return Value**
Type: Boolean

**getHistoricalSnapshotDates()**
Returns a list of historical snapshot dates.

**Syntax**
```
public LIST<String> getHistoricalSnapshotDates()
```

**Return Value**
Type: List<String>

**getId()**
Returns the unique report ID.

**Syntax**
```
public Id getId()
```
Return Value
Type: Id

getName()
Returns the report name.

Syntax
public String getName()

Return Value
Type: String

getReportBooleanFilter()
Returns logic to parse custom field filters. The value is null when filter logic is not specified.

Syntax
public String getReportBooleanFilter()

Return Value
Type: String

getReportFilters()
Returns a list of each custom filter in the report along with the field name, filter operator, and filter value.

Syntax
public LIST<Reports.ReportFilter> getReportFilters()

Return Value
Type: List<Reports.ReportFilter>

getReportFormat()
Returns the format of the report.

Syntax
public Reports.ReportFormat getReportFormat()

Return Value
Type: Reports.ReportFormat
Usage
This value can be:
- TABULAR
- SUMMARY
- MATRIX

getReportType()
Returns the unique API name and display name for the report type.

Syntax
```java
public Reports.ReportType getReportType()
```

Return Value
Type: Reports.ReportType

getScope()
Returns the API name for the scope defined for the report. Scope values depend on the report type.

Signature
```java
public String getScope()
```

Return Value
Type: String

getShowGrandTotal()
Indicates whether the report shows the grand total.

Signature
```java
public Boolean getShowGrandTotal()
```

Return Value
Type: Boolean

getShowSubtotals()
Indicates whether the report shows subtotals, such as column or row totals.

Signature
```java
public Boolean getShowSubtotals()
```
Return Value
Type: Boolean

getSortBy()
Returns the list of columns on which the report is sorted. Currently, you can sort on only one column.

Signature
public List<Reports.SortColumn> getSortBy()

Return Value
Type: List<Reports.SortColumn>

getStandardDateFilter()
Returns information about the standard date filter for the report, such as the start date, end date, date range, and date field API name.

Signature
public Reports.StandardDateFilter getStandardDateFilter()

Return Value
Type: Reports.StandardDateFilter

getStandardFilters()
Returns a list of standard filters for the report.

Signature
public List<Reports.StandardFilter> getStandardFilters()

Return Value
Type: List<Reports.StandardFilter>

getTopRows()
Returns information about a row limit filter, including the number of rows returned and the sort order.

Signature
public Reports.TopRows getTopRows()

Return Value
Type: Reports.TopRows
**setAggregates (aggregates)**
Sets unique identifiers for standard or custom summary formula fields in the report.

**Signature**

```java
public void setAggregates(List<String> aggregates)
```

**Parameters**

- `aggregates`
  - Type: List<String>

**Return Value**

Type: void

---

**setBuckets (buckets)**
Creates bucket fields in a report.

**Signature**

```java
public void setBuckets(List<Reports.BucketField> buckets)
```

**Parameters**

- `buckets`
  - Type: List<Reports.BucketField>

**Return Value**

Type: void

---

**setCrossFilters (crossFilters)**
Applies cross filters to a report.

**Signature**

```java
public void setCrossFilters(List<Reports.CrossFilter> crossFilters)
```

**Parameters**

- `crossFilter`
  - Type: List<Reports.CrossFilter>

**Return Value**

Type: void
**setCurrencyCode(currencyCode)**
Sets the currency, such as USD, EUR, or GBP, for report summary fields in an organization that has multicurrency enabled.

**Signature**

```java
public void setCurrencyCode(String currencyCode)
```

**Parameters**

- `currencyCode`
  Type: `String`

**Return Value**
Type: `void`

**setCustomSummaryFormula(customSummaryFormula)**
Adds a custom summary formula to a report.

**Signature**

```java
public void setCustomSummaryFormula(Map<String, Reports.ReportCsf> customSummaryFormula)
```

**Parameters**

- `customSummaryFormula`
  Type: `Map<String, Reports.ReportCsf>`

**Return Value**
Type: `void`

**setDescription(description)**
Sets the description of the report.

**Signature**

```java
public void setDescription(String description)
```

**Parameters**

- `description`
  Type: `String`

**Return Value**
Type: `void`
**setDetailColumns**
Sets the unique API names for the fields that contain detailed data—for example, OPPORTUNITY_NAME, TYPE, LEAD_SOURCE, or AMOUNT.

**Signature**
```java
public void setDetailColumns(List<String> detailColumns)
```

**Parameters**
- `detailColumns`
  - Type: List<String>

**Return Value**
Type: void

**setDeveloperName**
Sets the report API name—such as Closed_Sales_This_Quarter.

**Signature**
```java
public void setDeveloperName(String developerName)
```

**Parameters**
- `developerName`
  - Type: String

**setDivision**
Sets the division of the report.

---

**Note:** Reports that use standard filters (such as My Cases or My Team’s Accounts) show records in all divisions. These reports can’t be further limited to a specific division.

**Signature**
```java
public void setDivision(String division)
```

**Parameters**
- `division`
  - Type: String
Return Value
Type: void

setGroupingsAcross(groupingInfo)
Sets column groupings in a report.

Signature
public void setGroupingsAcross(List<Reports.GroupingInfo> groupingInfo)

Parameters

Parameters

groupingInfo
Type: List<Reports.GroupingInfo>

Return Value
Type: void

setGroupingsDown(groupingInfo)
Sets row groupings for a report.

Signature
public void setGroupingsDown(List<Reports.GroupingInfo> groupingInfo)

Parameters

groupingInfo
Type: List<Reports.GroupingInfo>

Return Value
Type: void

setHasDetailRows(hasDetailRows)
Specifies whether the report has detail rows.

Signature
public void setHasDetailRows(Boolean hasDetailRows)

Parameters

Parameters

hasDetailRows
Type: Boolean
Return Value
Type: void

**setHasRecordCount** (hasRecordCount)
Specifies whether the report is configured to show the total number of records.

**Signature**
public void setHasRecordCount(Boolean hasRecordCount)

**Parameters**
- hasRecordCount
  Type: Boolean

Return Value
Type: void

**setHistoricalSnapshotDates** (historicalSnapshot)
Sets a list of historical snapshot dates.

**Syntax**
public Void setHistoricalSnapshotDates(List<String> historicalSnapshot)

**Parameters**
- historicalSnapshot
  Type: List<String>

Return Value
Type: Void

**setId** (id)
Sets the unique report ID.

**Signature**
public void setId(Id id)

**Parameters**
- id
  Type: Id
Return Value
Type: void

**setName**(name)
Sets the report name.

Signature
```
public void setName(String name)
```

Parameters
- **name**
  Type: String

Return Value
Type: void

**setReportBooleanFilter**(reportBooleanFilter)
Sets logic to parse custom field filters.

Syntax
```
public Void setReportBooleanFilter(String reportBooleanFilter)
```

Parameters
- **reportBooleanFilter**
  Type: String

Return Value
Type: Void

**setReportFilters**(reportFilters)
Sets a list of each custom filter in the report along with the field name, filter operator, and filter value.

Syntax
```
public Void setReportFilters(LIST<Reports.ReportFilter> reportFilters)
```

Parameters
- **reportFilters**
  Type: List<Reports.ReportFilter>
Return Value
Type: Void

**setReportFormat(format)**
Sets the format of the report.

**Signature**
```java
public void setReportFormat(Reports.ReportFormat format)
```

**Parameters**
- `format`
  Type: `Reports.ReportFormat`

**Return Value**
Type: void

**setReportType(reportType)**
Sets the unique API name and display name for the report type.

**Signature**
```java
public void setReportType(Reports.ReportType reportType)
```

**Parameters**
- `reportType`
  Type: `Reports.ReportType`

**Return Value**
Type: void

**setScope(scopeName)**
Sets the API name for the scope defined for the report. Scope values depend on the report type.

**Signature**
```java
public void setScope(String scopeName)
```

**Parameters**
- `scopeName`
  Type: `String`
Return Value
Type: void

`setShowGrandTotal(showGrandTotal)`
Specifies whether the report shows the grand total.

**Signature**

```
public void setShowGrandTotal(Boolean showGrandTotal)
```

**Parameters**

`showGrandTotal`
Type: `Boolean`

Return Value
Type: void

`setShowSubtotals(showSubtotals)`
Specifies whether the report shows subtotals, such as column or row totals.

**Signature**

```
public void setShowSubtotals(Boolean showSubtotals)
```

**Parameters**

`showSubtotals`
Type: `Boolean`

Return Value
Type: void

`setSortBy(column)`
Sets the list of columns on which the report is sorted. Currently, you can only sort on one column.

**Signature**

```
public void setSortBy(List<Reports.SortColumn> column)
```

**Parameters**

`column`
Type: `List<Reports.SortColumn>`
Return Value
Type: void

`setStandardDateFilter(dateFilter)`
Sets the standard date filter—which includes the start date, end date, date range, and date field API name—for the report.

Signature
`public void setStandardDateFilter(Reports.StandardDateFilter dateFilter)`

Parameters
`dateFilter`  
Type: `Reports.StandardDateFilter`

Return Value
Type: void

`setStandardFilters(filters)`
Sets one or more standard filters on the report.

Signature
`public void setStandardFilters(List<Reports.StandardFilter> filters)`

Parameters
`filters`  
Type: `List<Reports.StandardFilter>`

Return Value
Type: void

`setTopRows(topRows)`
Applies a row limit filter to a report.

Signature
`public Reports.TopRows setTopRows(Reports.TopRows topRows)`

Parameters
`topRows`  
Type: `Reports.TopRows`
**Return Value**

Type: void

---

**ReportResults Class**

Contains the results of running a report.

---

**Namespace**

Reports

---

**ReportResults Methods**

The following are methods for ReportResults. All are instance methods.

---

**IN THIS SECTION:**

- **getAllData()**
  Returns all report data.
- **getFactMap()**
  Returns summary-level data or summary and detailed data for each row or column grouping. Detailed data is available if the includeDetails parameter is set to true when the report is run.
- **getGroupingsAcross()**
  Returns a collection of column groupings, keys, and values.
- **getGroupingsDown()**
  Returns a collection of row groupings, keys, and values.
- **getHasDetailRows()**
  Returns information about whether the fact map has detail rows.
- **getReportExtendedMetadata()**
  Returns additional, detailed metadata about the report, including data type and label information for groupings and summaries.
- **getReportMetadata()**
  Returns metadata about the report, including grouping and summary information.

---

**getAllData()**

Returns all report data.

**Syntax**

```java
public Boolean getAllData()
```

**Return Value**

Type: Boolean
Usage
When true, indicates that all report results are returned.
When false, indicates that results are returned for the same number of rows as in a report run in Salesforce.

Note: For reports that contain too many records, use filters to refine results.

getFactMap()
Returns summary-level data or summary and detailed data for each row or column grouping. Detailed data is available if the includeDetails parameter is set to true when the report is run.

Syntax
public MAP<String,Reports.ReportFact> getFactMap()

Return Value
Type: Map<String,Reports.ReportFact>

getGroupingsAcross()
Returns a collection of column groupings, keys, and values.

Syntax
public Reports.Dimension getGroupingsAcross()

Return Value
Type: Reports.Dimension

getGroupingsDown()
Returns a collection of row groupings, keys, and values.

Syntax
public Reports.Dimension getGroupingsDown()

Return Value
Type: Reports.Dimension

getHasDetailRows()
Returns information about whether the fact map has detail rows.

Syntax
public Boolean getHasDetailRows()
Return Value
Type: Boolean

Usage
• When true, indicates that the fact map returns values for summary-level and record-level data.
• When false, indicates that the fact map returns summary values.

getReportExtendedMetadata() (getReportExtendedMetadata)
Returns additional, detailed metadata about the report, including data type and label information for groupings and summaries.

Syntax
public Reports.ReportExtendedMetadata getReportExtendedMetadata()

Return Value
Type: Reports.ReportExtendedMetadata

getReportMetadata() (getReportMetadata)
Returns metadata about the report, including grouping and summary information.

Syntax
public Reports.ReportMetadata getReportMetadata()

Return Value
Type: Reports.ReportMetadata

ReportScopeInfo Class
Contains information about possible scope values that you can choose. Scope values depend on the report type. For example, you can set the scope for opportunity reports to All opportunities, My team’s opportunities, or My opportunities.

Namespace
Reports

IN THIS SECTION:
  ReportScopeInfo Methods

ReportScopeInfo Methods
The following are methods for ReportScopeInfo.
IN THIS SECTION:

getDefaultValue()
Returns the default scope of the data to display in the report.

getValues()
Returns a list of scope values specified for the report.

**getDefaultValue()**

Returns the default scope of the data to display in the report.

**Signature**

```java
public String getDefaultValue()
```

**Return Value**

Type: `String`

**getValues()**

Returns a list of scope values specified for the report.

**Signature**

```java
public List<Reports.ReportScopeValue> getValues()
```

**Return Value**

Type: `List<Reports.ReportScopeValue>`

**ReportScopeValue Class**

Contains information about a possible scope value. Scope values depend on the report type. For example, you can set the scope for opportunity reports to **All opportunities**, **My team’s opportunities**, or **My opportunities**.

**Namespace**

```java
Reports
```

**ReportScopeValue Methods**

The following are methods for `ReportScopeValue`. 
IN THIS SECTION:
  - `getAllowsDivision()`
    Returns a boolean value that indicates whether you can segment the report by this scope.
  - `getLabel()`
    Returns the display name of the scope of the report.
  - `getValue()`
    Returns the scope value for the report.

`getAllowsDivision()`
Returns a boolean value that indicates whether you can segment the report by this scope.

**Signature**

```java
public Boolean getAllowsDivision()
```

**Return Value**

Type: `Boolean`

`getLabel()`
Returns the display name of the scope of the report.

**Signature**

```java
public String getLabel()
```

**Return Value**

Type: `String`

`getValue()`
Returns the scope value for the report.

**Signature**

```java
public String getValue()
```

**Return Value**

Type: `String`

**ReportType Class**

Contains the unique API name and display name for the report type.
Namespace

Reports

ReportType Methods

The following are methods for ReportType. All are instance methods.

IN THIS SECTION:

getLabel()  
Returns the localized display name of the report type.

type()  
Returns the unique identifier of the report type.

gGetLabel()  
Returns the localized display name of the report type.

Syntax

public String getLabel()

Return Value

Type: String

getType()  
Returns the unique identifier of the report type.

gType()  
Returns the unique identifier of the report type.

Syntax

public String getType()

Return Value

Type: String

ReportTypeColumn Class

Contains detailed report type metadata about a field, including data type, display name, and filter values.

Namespace

Reports
ReportTypeColumn Methods

The following are methods for ReportTypeColumn. All are instance methods.

IN THIS SECTION:
  - getDataType()
  - getFilterValues()
  - getFilterable()
  - getLabel()
  - getName()

**getDataType()**

Returns the data type of the field.

**Syntax**

```java
public Reports.ColumnDataType getDataType()
```

**Return Value**

Type: `Reports.ColumnDataType`

**getFilterValues()**

If the field data type is picklist, multi-select picklist, boolean, or checkbox, returns all filter values for a field. For example, checkbox fields always have a value of `true` or `false`. For fields of other data types, the filter value is an empty array, because their values can’t be determined.

**Syntax**

```java
public LIST<Reports.FilterValue> getFilterValues()
```

**Return Value**

Type: `List<Reports.FilterValue>`

**getFilterable()**

If the field is of a type that can’t be filtered, returns `False`. For example, fields of the type `Encrypted Text` can’t be filtered.

**getLabel()**

Returns the localized display name of the field.

**getName()**

Returns the unique API name of the field.
ReportTypeColumnCategory Class

Information about categories of fields in a report type.

Namespace

Reports

Usage

A report type column category is a set of fields that the report type grants access to. For example, an opportunity report has categories like Opportunity Information and Primary Contact. The Opportunity Information category has fields like Amount, Probability, and Close Date.

Get category information about a report by first getting the report metadata:

```java
// Get the report ID
List<Report> reportList = [SELECT Id, DeveloperName FROM Report where DeveloperName = 'Q1_Opportunities2'];
```
String reportId = (String)reportList.get(0).get('Id');

// Describe the report
Reports.ReportDescribeResult describeResults =

Reports.ReportManager.describeReport(reportId);

// Get report type metadata
Reports.ReportTypeMetadata reportTypeMetadata = describeResults.getReportTypeMetadata();

// Get report type column categories
List<Reports.ReportTypeColumnCategory> reportTypeColumnCategories =

reportTypeMetadata.getCategories();

System.debug('reportTypeColumnCategories: ' + reportTypeColumnCategories);

### ReportTypeColumnCategory Methods

The following are methods for `ReportTypeColumnCategory`. All are instance methods.

**IN THIS SECTION:**

- **getColumn()**
  
  Returns information for all fields in the report type. The information is organized by each section’s unique API name.

- **getLabel()**
  
  Returns the localized display name of a section in the report type under which fields are organized. For example, in an Accounts with Contacts custom report type, **Account General** is the display name of the section that contains fields on general account information.

**getColumn()**

Returns information for all fields in the report type. The information is organized by each section’s unique API name.

**Syntax**

```java
public Map<String, Reports.ReportTypeColumn> getColumn()
```

**Return Value**

Type: `Map<String, Reports.ReportTypeColumn>`

**getLabel()**

Returns the localized display name of a section in the report type under which fields are organized. For example, in an Accounts with Contacts custom report type, **Account General** is the display name of the section that contains fields on general account information.

**Syntax**

```java
public String getLabel()
```
ReportTypeMetadata Class
Contains report type metadata, which gives you information about the fields that are available in each section of the report type, plus filter information for those fields.

Namespace
Reports

IN THIS SECTION:
ReportTypeMetadata Methods

ReportTypeMetadata Methods
The following are methods for ReportTypeMetadata. All are instance methods.

IN THIS SECTION:
getCategories() Returns all fields in the report type. The fields are organized by section.
getDivisionInfo() Returns the default division and a list of all possible divisions that can be applied to this type of report.
getScopeInfo() Returns information about the scopes that can be applied to this type of report.
getStandardDateFilterDurationGroups() Returns information about the standard date filter groupings that can be applied to this type of report. Standard date filter groupings include Calendar Year, Calendar Quarter, Calendar Month, Calendar Week, Fiscal Year, Fiscal Quarter, Day and a custom value based on a user-defined date range.
getStandardFilterInfos() Returns information about standard date filters that can be applied to this type of report.

getCategories()
Returns all fields in the report type. The fields are organized by section.

Syntax
public LIST<Reports.ReportTypeColumnCategory> getCategories()

Return Value
Type: List<Reports.ReportTypeColumnCategory>
getDivisionInfo()

Returns the default division and a list of all possible divisions that can be applied to this type of report.

Signature

```java
public Reports.ReportDivisionInfo getDivisionInfo()
```

Return Value

Type: Reports.ReportDivisionInfo

getScopeInfo()

Returns information about the scopes that can be applied to this type of report.

Signature

```java
public Reports.ReportScopeInfo getScopeInfo()
```

Return Value

Type: Reports.ReportScopeInfo

getStandardDateFilterDurationGroups()

Returns information about the standard date filter groupings that can be applied to this type of report. Standard date filter groupings include Calendar Year, Calendar Quarter, Calendar Month, Calendar Week, Fiscal Year, Fiscal Quarter, Day and a custom value based on a user-defined date range.

Signature

```java
public List<Reports.StandardDateFilterDurationGroup> getStandardDateFilterDurationGroups()
```

Return Value

Type: List<Reports.StandardDateFilterDurationGroup>

getStandardFilterInfos()

Returns information about standard date filters that can be applied to this type of report.

Signature

```java
public Map<String,Reports.StandardFilterInfo> getStandardFilterInfos()
```

Return Value

Type: Map<String,Reports.StandardFilterInfo>
SortColumn Class
Contains information about the sort column used in the report.

Namespace
Reports

IN THIS SECTION:
SortColumn Methods

SortColumn Methods
The following are methods for SortColumn.

IN THIS SECTION:
getSortColumn()
Returns the column used to sort the records in the report.

getSortOrder()
Returns the sort order—ascending or descending—for the sort column.

setSortColumn(sortColumn)
Sets the column used to sort the records in the report.

setSortOrder(SortOrder)
Sets the sort order—ascending or descending—for the sort column.

getSortColumn()
Returns the column used to sort the records in the report.

Signature
public String getSortColumn()

Return Value
Type: String

getSortOrder()
Returns the sort order—ascending or descending—for the sort column.

Signature
public Reports.ColumnSortOrder getSortOrder()
Return Value
Type: Reports.ColumnSortOrder

`setSortColumn(sortColumn)`
Sets the column used to sort the records in the report.

Signature
`public void setSortColumn(String sortColumn)`

Parameters
`sortColumn`
Type: String

Return Value
Type: void

`setSortOrder(SortOrder)`
Sets the sort order—ascending or descending—for the sort column.

Signature
`public void setSortOrder(Reports.ColumnSortOrder sortOrder)`

Parameters
`sortOrder`
Type: Reports.ColumnSortOrder

Return Value
Type: void

**StandardDateFilter Class**
Contains information about standard date filter available in the report—for example, the API name, start date, and end date of the standard date filter duration as well as the API name of the date field on which the filter is placed.

**Namespace**
Reports

IN THIS SECTION:
StandardDateFilter Methods
StandardDateFilter Methods

The following are methods for StandardDateFilter.

IN THIS SECTION:

- **getColumn()**
  Returns the API name of the standard date filter column.

- **getDurationValue()**
  Returns duration information about a standard date filter, such as start date, end date, and display name and API name of the date filter.

- **getEndDate()**
  Returns the end date of the standard date filter.

- **getStartDate()**
  Returns the start date for the standard date filter.

- **setColumn(standardDateFilterColumnName)**
  Sets the API name of the standard date filter column.

- **setDurationValue(durationName)**
  Sets the API name of the standard date filter.

- **setEndDate(endDate)**
  Sets the end date for the standard date filter.

- **setStartDate(startDate)**
  Sets the start date for the standard date filter.

### getColumn()

Returns the API name of the standard date filter column.

**Signature**

```java
public String getColumn()
```

**Return Value**

Type: String

### getDurationValue()

Returns duration information about a standard date filter, such as start date, end date, and display name and API name of the date filter.

**Signature**

```java
public String getDurationValue()
```

**Return Value**

Type: String
**getEndDate()**
Returns the end date of the standard date filter.

**Signature**
```java
class StandardDateFilter {
    public String getEndDate()
}
```

**Return Value**
Type: String

**getStartDate()**
Returns the start date for the standard date filter.

**Signature**
```java
class StandardDateFilter {
    public String getStartDate()
}
```

**Return Value**
Type: String

**setColumn(standardDateFilterColumnName)**
Sets the API name of the standard date filter column.

**Signature**
```java
class StandardDateFilter {
    public void setColumn(String standardDateFilterColumnName)
}
```

**Parameters**

*standardDateFilterColumnName*  
Type: String

**Return Value**
Type: void

**setDurationValue(durationName)**
Sets the API name of the standard date filter.

**Signature**
```java
class StandardDateFilter {
    public void setDurationValue(String durationName)
}
```
Parameters

durationName
Type: String

Return Value
Type: void

setEndDate(endDate)
Sets the end date for the standard date filter.

Signature

public void setEndDate(String endDate)

Parameters

date
Type: String

Return Value
Type: void

setStartDate(startDate)
Sets the start date for the standard date filter.

Signature

public void setStartDate(String startDate)

Parameters

startDate
Type: String

Return Value
Type: void

StandardDateFilterDuration Class

Contains information about each standard date filter—also referred to as a relative date filter. It contains the API name and display label of the standard date filter duration as well as the start and end dates.

Namespace

Reports
StandardDateFilterDuration Methods

The following are methods for StandardDateFilterDuration.

getEndDate()
Returns the end date of the date filter.

getLabel()
Returns the display name of the date filter. Possible values are relative date filters—like Current FY and Current FQ—and custom date filters.

geteStartDate()
Returns the start date of the date filter.

getValue()
Returns the API name of the date filter. Possible values are relative date filters—like THIS_FISCAL_YEAR and NEXT_FISCAL_QUARTER—and custom date filters.

getEndDate()
Returns the end date of the date filter.

**Signature**

```java
public String getEndDate()
```

**Return Value**

Type: String

getLabel()
Returns the display name of the date filter. Possible values are relative date filters—like Current FY and Current FQ—and custom date filters.

**Signature**

```java
public String getLabel()
```

**Return Value**

Type: String

getStartDate()
Returns the start date of the date filter.
Signature

`public String getStartDate()`

Return Value

Type: `String`

getValue()

Returns the API name of the date filter. Possible values are relative date filters—like `THIS_FISCAL_YEAR` and `NEXT_FISCAL_QUARTER`—and custom date filters.

Signature

`public String getValue()`

Return Value

Type: `String`

**StandardDateFilterDurationGroup Class**

Contains information about the standard date filter groupings, such as the grouping display label and all standard date filters that fall under the grouping. Groupings include Calendar Year, Calendar Quarter, Calendar Month, Calendar Week, Fiscal Year, Fiscal Quarter, Day, and custom values based on user-defined date ranges.

**Namespace**

Reports

**IN THIS SECTION:**

**StandardDateFilterDurationGroup Methods**

The following are methods for `StandardDateFilterDurationGroup`.

**IN THIS SECTION:**

`getLabel()`

Returns the display label for the standard date filter grouping.

`getStandardDateFilterDurations()`

Returns the standard date filter groupings.

`getLabel()`

Returns the display label for the standard date filter grouping.
Signature

public String getLabel()

Return Value

Type: String

getStandardDateFilterDurations()

Returns the standard date filter groupings.

Signature

public List<Reports.StandardDateFilterDuration> getStandardDateFilterDurations()

Return Value

Type: List<Reports.StandardDateFilterDuration>

For example, a standard filter date grouping might look like this:

```
Reports.StandardDateFilterDuration[endDate=2015-12-31, label=Current FY, startDate=2015-01-01, value=THIS_FISCAL_YEAR],
Reports.StandardDateFilterDuration[endDate=2014-12-31, label=Previous FY, startDate=2014-01-01, value=LAST_FISCAL_YEAR],
Reports.StandardDateFilterDuration[endDate=2014-12-31, label=Previous 2 FY, startDate=2013-01-01, value=LAST_N_FISCAL_YEARS:2]
```

StandardFilter Class

Contains information about the standard filter defined in the report, such as the filter field API name and filter value.

Namespace

Reports

Usage

Use to get or set standard filters on a report. Standard filters vary by report type. For example, standard filters for reports on the Opportunity object are Show, Opportunity Status, and Probability.

IN THIS SECTION:

StandardFilter Methods

StandardFilter Methods

The following are methods for StandardFilter.
IN THIS SECTION:

getName()  
Return the API name of the standard filter.

getValue()  
Returns the standard filter value.

setName(name)  
Sets the API name of the standard filter.

setValue(value)  
Sets the standard filter value.

**getName()**  
Return the API name of the standard filter.

**Signature**  
public String getName()  

**Return Value**  
Type: String

**getValue()**  
Returns the standard filter value.

**Signature**  
public String getValue()  

**Return Value**  
Type: String

**setName(name)**  
Sets the API name of the standard filter.

**Signature**  
public void setName(String name)  

**Parameters**  

name  
Type: String
Return Value
Type: void

`setValue(value)`
Sets the standard filter value.

Signature
`public void setValue(String value)`

Parameters
`value`
Type: `String`

Return Value
Type: void

**StandardFilterInfo Class**
Is an abstract base class for an object that provides standard filter information.

**Namespace**
*Reports*

IN THIS SECTION:
- StandardFilterInfo Methods

**StandardFilterInfo Methods**
The following are methods for `StandardFilterInfo`.

IN THIS SECTION:
- `getLabel()`
  Returns the display label of the standard filter.
- `getType()`
  Returns the type of standard filter.

`getLabel()`
Returns the display label of the standard filter.
StandardFilterInfoPicklist Class

Contains information about the standard filter picklist, such as the display name and type of the filter field, the default picklist value, and a list of all possible picklist values.

Namespace

Reports

IN THIS SECTION:

StandardFilterInfoPicklist Methods

The following are methods for StandardFilterInfoPicklist.

IN THIS SECTION:

getDefaultValue()
Returns the default value for the standard filter picklist.

getFilterValues()
Returns a list of standard filter picklist values.

getLabel()
Returns the display name of the standard filter picklist.

getType()
Returns the type of the standard filter picklist.
**getDefaultValue()**
Returns the default value for the standard filter picklist.

**Signature**
```
public String getDefaultValue()
```

**Return Value**
Type: `String`

**getFilterValues()**
Returns a list of standard filter picklist values.

**Signature**
```
public List<Reports.FilterValue> getFilterValues()
```

**Return Value**
Type: `List<Reports.FilterValue>`

**getLabel()**
Returns the display name of the standard filter picklist.

**Signature**
```
public String getLabel()
```

**Return Value**
Type: `String`

**getType()**
Returns the type of the standard filter picklist.

**Signature**
```
public Reports.StandardFilterType getType()
```

**Return Value**
Type: `Reports.StandardFilterType`
StandardFilterType Enum

The StandardFilterType enum describes the type of standard filters in a report. The `getType()` method returns a `Reports.StandardFilterType` enum value.

Namespace

Reports

Enum Values

The following are the values of the `Reports.StandardFilterType` enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PICKLIST</td>
<td>Values for the standard filter type.</td>
</tr>
<tr>
<td>STRING</td>
<td>String values.</td>
</tr>
</tbody>
</table>

SummaryValue Class

Contains summary data for a cell of the report.

Namespace

Reports

SummaryValue Methods

The following are methods for `SummaryValue`. All are instance methods.

IN THIS SECTION:

getLabel()

Returns the formatted summary data for a specified cell.

getValue()

Returns the numeric value of the summary data for a specified cell.

getLabel ()

Returns the formatted summary data for a specified cell.

Syntax

```java
public String getLabel()
```

Return Value

Type: `String`
getValue()  
Returns the numeric value of the summary data for a specified cell.

Syntax  
public Object getValue()  

Return Value  
Type: Object

ThresholdInformation Class  
Contains a list of evaluated conditions for a report notification.

Namespace  
Reports

IN THIS SECTION:  
ThresholdInformation Constructors  
ThresholdInformation Methods

ThresholdInformation Constructors  
The following are constructors for ThresholdInformation.

IN THIS SECTION:  
ThresholdInformation(evaluatedConditions)  
Creates a new instance of the Reports.EvaluatedCondition class.

ThresholdInformation (evaluatedConditions)  
Creates a new instance of the Reports.EvaluatedCondition class.

Signature  
public ThresholdInformation(List<Reports.EvaluatedCondition> evaluatedConditions)

Parameters  
evaluatedConditions  
Type: List<Reports.EvaluatedCondition>  
A list of Reports.EvaluatedCondition objects.
ThresholdInformation Methods

The following are methods for ThresholdInformation.

IN THIS SECTION:

getEvaluatedConditions()
Returns a list of evaluated conditions for a report notification.

getEvaluatedConditions()
Returns a list of evaluated conditions for a report notification.

Signature

public List<Reports.EvaluatedCondition> getEvaluatedConditions()

Return Value

Type: List<Reports.EvaluatedCondition>

TopRows Class

Contains methods and constructors for working with information about a row limit filter.

Namespace

Reports

IN THIS SECTION:

TopRows Constructors
TopRows Methods

TopRows Constructors

The following are constructors for TopRows.

IN THIS SECTION:

TopRows(rowLimit, direction)
Creates an instance of the Reports.TopRows class using the specified parameters.

TopRows()
Creates an instance of the Reports.TopRows class. You can then set values by using the class’s set methods.

TopRows (rowLimit, direction)
Creates an instance of the Reports.TopRows class using the specified parameters.
Signature

```java
public TopRows(Integer rowLimit, Reports.ColumnSortOrder direction)
```

Parameters

- **rowLimit**
  - Type: `Integer`
  - The number of rows returned in the report.

- **direction**
  - Type: `Reports.ColumnSortOrder`
  - The sort order of the report rows.

**TopRows()**

Creates an instance of the `Reports.TopRows` class. You can then set values by using the class's `set` methods.

Signature

```java
public TopRows()
```

**TopRows Methods**

The following are methods for `TopRows`.

In this section:

- `getDirection()`: Returns the sort order of the report rows.
- `getRowLimit()`: Returns the maximum number of rows shown in the report.
- `setDirection(value)`: Sets the sort order of the report's rows.
- `setDirection(direction)`: Sets the sort order of the report's rows.
- `setRowLimit(rowLimit)`: Sets the maximum number of rows included in the report.
- `toString()`: Returns a string.

**getDirection()**

Returns the sort order of the report rows.

Signature

```java
public Reports.ColumnSortOrder getDirection()
```
Return Value
Type: Reports.ColumnSortOrder

getRowLimit()
Returns the maximum number of rows shown in the report.

Signature
public Integer getRowLimit()

Return Value
Type: Integer

setDirection(value)
Sets the sort order of the report’s rows.

Signature
public void setDirection(String value)

Parameters
value
Type: String
For possible values, see Reports.ColumnSortOrder.

Return Value
Type: void

setDirection(direction)
Sets the sort order of the report’s rows.

Signature
public void setDirection(Reports.ColumnSortOrder direction)

Parameters
direction
Type: Reports.ColumnSortOrder

Return Value
Type: void
setRowLimit(rowLimit)
Sets the maximum number of rows included in the report.

Signature
public void setRowLimit(Integer rowLimit)

Parameters
rowLimit
Type: Integer

Return Value
Type: void

toString()
Returns a string.

Signature
public String toString()

Return Value
Type: String

Reports Exceptions
The Reports namespace contains exception classes.
All exception classes support built-in methods for returning the error message and exception type. See Exception Class and Built-In Exceptions on page 3076.
The Reports namespace contains these exceptions:

<table>
<thead>
<tr>
<th>Exception</th>
<th>Description</th>
<th>Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reports.FeatureNotSupportedException</td>
<td>Invalid report format</td>
<td></td>
</tr>
<tr>
<td>Reports.InstanceAccessException</td>
<td>Unable to access report instance</td>
<td></td>
</tr>
<tr>
<td>Reports.InvalidFilterException</td>
<td>Filter validation error</td>
<td>List&lt;String&gt; getFilterErrors() returns a list of filter errors</td>
</tr>
<tr>
<td>Reports.InvalidReportMetadataException</td>
<td>Missing metadata for filters</td>
<td>List&lt;String&gt; getReportMetadataErrors() returns a list of metadata errors</td>
</tr>
<tr>
<td>Reports.InvalidSnapshotDateException</td>
<td>Invalid historical report format</td>
<td>List&lt;String&gt; getSnapshotDateErrors() returns a list of snapshot date errors</td>
</tr>
</tbody>
</table>
RichMessaging Namespace

Provides objects and methods for handling content in enhanced Messaging channels.

The following are the classes in the RichMessaging namespace.

IN THIS SECTION:

AbstractTiming Class
Parent class for other RichMessaging timing classes.

AddressableContact Class
Represents an addressable contact.

AuthRequestHandler Interface
Use this interface to handle authorization request responses.

AuthRequestResponse Class
This class contains authorization request response data.

AuthRequestResult Class
This class contains the result from handling the authorization request response.

AuthRequestResultStatus Enum
This enum describes the authentication result status.

DeferredTiming Class
Represents timing for a transaction that occurs in the future.

MessageDefinitionInputParameter Class
Represents a messaging component parameter value. This class is used to provide parameter payloads that can be translated to structured content payloads in rich content messages.

PaymentItemStatus Enum
Represents the status of a payment item in payment requests sent in enhanced Messaging channels.

PaymentLineItem Class
Represents a payment line item in payment requests sent in enhanced Messaging channels.

PaymentMethod Class
Represents a payment method.

PostalAddress Class
Represents the postal address.
ProcessPaymentHandler Interface
Interface used to process payment requests.

ProcessPaymentRequest Class
Represents a request to process a payment.

ProcessPaymentResult Class
Represents the result of a payment processing operation.

ProcessPaymentResultStatus Enum
Represents the status of a payment processing result.

RecurringTiming Class
Represents a payment that occurs on a regular basis.

ShippingMethod Class
Represents a shipping method listed in payment requests sent in enhanced Messaging channels.

TimeSlotOption Class
Represents a complex time slot option type. This class is used to provide time option payloads that can be translated to structured content payloads in rich content messages.

TimeIntervalUnit Enum
Represents an enumerated type that describes the timing interval.

TimingType Enum
Represents an enumerated type that describes the type of timing.

AbstractTiming Class
Parent class for other RichMessaging timing classes.

Namespace
RichMessaging

SEE ALSO:
DeferredTiming Class
RecurringTiming Class

AddressableContact Class
Represents an addressable contact.

Namespace
RichMessaging

IN THIS SECTION:
AddressableContact Constructors
AddressableContact Properties
AddressableContact Constructors

The following are constructors for AddressableContact.

IN THIS SECTION:

AddressableContact(givenName, phoneticGivenName, familyName, phoneticFamilyName, emailAddress, phoneNumber, postalAddress)

Creates a new instance of the RichMessaging.AddressableContact class.

AddressableContact(givenName, phoneticGivenName, familyName, phoneticFamilyName, emailAddress, phoneNumber, postalAddress)

Creates a new instance of the RichMessaging.AddressableContact class.

Signature

public AddressableContact(String givenName, String phoneticGivenName, String familyName, String phoneticFamilyName, String emailAddress, String phoneNumber, RichMessaging.PostalAddress postalAddress)

Parameters

givenName
Type: String
The contact’s first name.

phoneticGivenName
Type: String
The phonetic spelling of the contact’s first name.

familyName
Type: String
The contact’s surname.

phoneticFamilyName
Type: String
The phonetic spelling of the contact’s surname.

emailAddress
Type: String
The contact’s email address.

phoneNumber
Type: String
The contact’s phone number.

postalAddress
Type: RichMessaging.PostalAddress
The contact’s postal address.
AddressableContact Properties

The following are properties for AddressableContact.

IN THIS SECTION:

- **emailAddress**
  The contact’s email address.

- **familyName**
  The contact’s surname.

- **givenName**
  The contact’s first name.

- **phoneNumber**
  The contact’s phone number.

- **phoneticFamilyName**
  The phonetic spelling of the contact’s surname.

- **phoneticGivenName**
  The phonetic spelling of the contact’s first name.

- **postalAddress**
  The contact’s postal address.

**emailAddress**

The contact’s email address.

**Signature**

```csharp
public String emailAddress {get; set;}
```

**Property Value**

Type: String

**familyName**

The contact’s surname.

**Signature**

```csharp
public String familyName {get; set;}
```

**Property Value**

Type: String

**givenName**

The contact’s first name.

**Signature**

```csharp
public String givenName {get; set;}
```

**Property Value**

Type: String
Signature
public String givenName {get; set;}

Property Value
Type: String

givenName
The contact’s given name.

Signature
public String givenName {get; set;}

Property Value
Type: String

givenName
The contact’s given name.

Signature
public String givenName {get; set;}

Property Value
Type: String

givenName
The contact’s given name.

Signature
public String givenName {get; set;}

Property Value
Type: String

givenName
The contact’s given name.

Signature
public String givenName {get; set;}

Property Value
Type: String

givenName
The contact’s given name.

Signature
public String givenName {get; set;}

Property Value
Type: String

givenName
The contact’s given name.

Signature
public String givenName {get; set;}

Property Value
Type: String

givenName
The contact’s given name.

Signature
public String givenName {get; set;}

Property Value
Type: String

givenName
The contact’s given name.

Signature
public String givenName {get; set;}

Property Value
Type: String

givenName
The contact’s given name.

Signature
public String givenName {get; set;}

Property Value
Type: String

givenName
The contact’s given name.

Signature
public String givenName {get; set;}

Property Value
Type: String

givenName
The contact’s given name.

Signature
public String givenName {get; set;}

Property Value
Type: String

givenName
The contact’s given name.

Signature
public String givenName {get; set;}

Property Value
Type: String

givenName
The contact’s given name.

Signature
public String givenName {get; set;}

Property Value
Type: String

givenName
The contact’s given name.

Signature
public String givenName {get; set;}

Property Value
Type: String

givenName
The contact’s given name.

Signature
public String givenName {get; set;}

Property Value
Type: String

givenName
The contact’s given name.

Signature
public String givenName {get; set;}

Property Value
Type: String

givenName
The contact’s given name.

Signature
public String givenName {get; set;}

Property Value
Type: String

givenName
The contact’s given name.

Signature
public String givenName {get; set;}

Property Value
Type: String

givenName
The contact’s given name.
Property Value
Type: RichMessaging.PostalAddress

AuthRequestHandler Interface
Use this interface to handle authorization request responses.

Namespace
RichMessaging on page 2653

Usage
When using this interface, the following limits are overridden. See Execution Governors and Limits in the Apex Developer Guide.

Table 1: Overridden Limits

<table>
<thead>
<tr>
<th>Limit Name</th>
<th>Overridden Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total number of SOQL queries issued</td>
<td>2</td>
</tr>
<tr>
<td>Total number of records retrieved by a single SOSL query</td>
<td>2</td>
</tr>
<tr>
<td>Total number of DML statements issued</td>
<td>1</td>
</tr>
<tr>
<td>Total number of records processed as a result of DML statements</td>
<td>1</td>
</tr>
<tr>
<td>Total number of callouts (HTTP requests or web services calls) in a transaction</td>
<td>2</td>
</tr>
</tbody>
</table>

IN THIS SECTION:
AuthRequestHandler Methods
AuthRequestHandler Example Implementation

AuthRequestHandler Methods
The following are methods for AuthRequestHandler.

IN THIS SECTION:
handleAuthRequest(var1)
Handles authorization request response.

handleAuthRequest(var1)
Handles authorization request response.
Signature

public RichMessaging.AuthRequestResult
handleAuthRequest(RichMessaging.AuthRequestResponse var1)

Parameters

var1
  Type: RichMessaging.AuthRequestResponse on page 2660
  The authorization response.

Return Value

Type: RichMessaging.AuthRequestResult on page 2662

AuthRequestHandler Example Implementation

This is an example implementation of the RichMessaging.AuthRequestHandler interface.

global class SampleAuthRequestHandler implements RichMessaging.AuthRequestHandler {
    global RichMessaging.AuthRequestResult
    handleAuthRequest(RichMessaging.AuthRequestResponse authReqResponse) {

        // Get contact email from messaging session
        String sessionId = authReqResponse.getContextRecordId();
        String contactEmail = [select MessagingSession.EndUserContact.Email from MessagingSession where id = :sessionId].EndUserContact.Email;

        DateTime dt = DateTime.now();

        // Get user info if there's a valid contact email
        if (!String.isBlank(contactEmail)) {
            String userInfoUrl = 'https://api.MY_AUTH_DOMAIN.com/v1/';

            HttpRequest req = new HttpRequest();
            req.setEndpoint(userInfoUrl);
            req.setHeader('Content-Type', 'application/json');
            req.setMethod('GET');
            req.setHeader('Authorization', 'Bearer ' + authReqResponse.getAccessToken());
            Http http = new Http();
            HTTPResponse res = http.send(req);

            String responseBody = res.getBody();
            UserWrapper userInfo = (UserWrapper)System.JSON.deserialize(responseBody, UserWrapper.class);

            if (userInfo.email == contactEmail) {
                authRequestStatus = RichMessaging.AuthRequestResultStatus.AUTHENTICATED;
            }
        }
    }
}
AuthRequestResponse Class

This class contains authorization request response data.

Namespace

RichMessaging

IN THIS SECTION:

AuthRequestResponse Constructors
AuthRequestResponse Methods

AuthRequestResponse Constructors

The following are constructors for AuthRequestResponse.

IN THIS SECTION:

AuthRequestResponse(accessToken, contextRecordId, authProviderName)

AuthRequestResponse(accessToken, contextRecordId, authProviderName)


Signature

public AuthRequestResponse(String accessToken, String contextRecordId, String authProviderName)
Parameters

accessToken
Type: String
The authorization access token.

contextRecordId
Type: String
The context record ID.

authProviderName
Type: String
The provider name.

AuthRequestResponse Methods

The following are methods for AuthRequestResponse.

IN THIS SECTION:

getAccessToken()
Gets the authorization access token.

getAuthProviderName()
Get the authorization provider name.

getContextRecordId()
Gets the context record ID.

getAccessToken ()

Gets the authorization access token.

Signature

public String getAccessToken()  

Return Value

Type: String
The access token.

getAuthProviderName ()

Get the authorization provider name.

Signature

public String getAuthProviderName()
Return Value
Type: String
The authorization provider name.

getContextRecordId()
Gets the context record ID.

Signature
public String getContextRecordId()

Return Value
Type: String
The context record ID.

AuthRequestResult Class
This class contains the result from handling the authorization request response.

Namespace
RichMessaging

IN THIS SECTION:

AuthRequestResult Constructors
AuthRequestResult Properties

AuthRequestResult Constructors
The following are constructors for AuthRequestResult.

IN THIS SECTION:

AuthRequestResult(redirectPageReference, resultStatus, expirationDateTime)

AuthRequestResult(redirectPageReference, resultStatus, expirationDateTime)

Signature
public AuthRequestResult(System.PageReference redirectPageReference, RichMessaging.AuthRequestResultStatus resultStatus, Datetime expirationDateTime)
Parameters

`redirectPageReference`
Type: `System.PageReference` on page 3268
The reference to the redirect page.

`resultStatus`
Type: `RichMessaging.AuthRequestResultStatus` on page 2664
The result status value.

`expirationDateTime`
Type: `Datetime`
The expiration time.

AuthRequestResult Properties

The following are properties for AuthRequestResult.

IN THIS SECTION:

`expirationDateTime`
The expiration date and time.

`redirectPageReference`
The reference to the redirect page.

`resultStatus`
The result status value.

`expirationDateTime`
The expiration date and time.

Signature

```
public Datetime expirationDateTime {get; set;}
```

Property Value
Type: `Datetime`

`redirectPageReference`
The reference to the redirect page.

Signature

```
public System.PageReference redirectPageReference {get; set;}
```

Property Value
Type: `System.PageReference` on page 3268
**resultStatus**
The result status value.

**Signature**
public RichMessaging.AuthRequestResultStatus resultStatus {get; set;}

**Property Value**
Type: RichMessaging.AuthRequestResultStatus on page 2664

**AuthRequestResultStatus Enum**
This enum describes the authentication result status.

**Enum Values**
The following are the values of the RichMessaging.AuthRequestResultStatus enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTHENTICATED</td>
<td>Authenticated result.</td>
</tr>
<tr>
<td>DECLINED</td>
<td>Declined result.</td>
</tr>
</tbody>
</table>

**DeferredTiming Class**
Represents timing for a transaction that occurs in the future.

**Namespace**
RichMessaging

**IN THIS SECTION:**
- DeferredTiming Constructors
- DeferredTiming Properties

**DeferredTiming Constructors**
The following are constructors for DeferredTiming.

**IN THIS SECTION:**
- DeferredTiming(deferredDate)
- DeferredTiming()
**DeferredTiming (deferredDate)**

**Signature**
public DeferredTiming(Datetime deferredDate)

**Parameters**
*deferredDate*
- Type: Datetime
  - The deferred date.

**DeferredTiming ()**

**Signature**
public DeferredTiming()

**DeferredTiming Properties**
The following are properties for DeferredTiming.

**IN THIS SECTION:**
- deferredDate
  - The deferred date. Invocable variable.
- deferredDateValue
  - The deferred date. Enabled for Lightning components.
- timingType
  - Always returns “DeferredTiming”.

**deferredDate**
The deferred date. Invocable variable.

**Signature**
public Datetime deferredDate {get; set;}

**Property Value**
Type: Datetime
**deferredDateValue**

The deferred date. Enabled for Lightning components.

**Signature**

public Datetime deferredDateValue {get; set;}

**Property Value**

Type: Datetime

**timingType**

Always returns “DeferredTiming”.

**Signature**

public String timingType {get; set;}

**Property Value**

Type: String

**MessageDefinitionInputParameter Class**

Represents a messaging component parameter value. This class is used to provide parameter payloads that can be translated to structured content payloads in rich content messages.

**Namespace**

RichMessaging

IN THIS SECTION:

MessageDefinitionInputParameter Properties

**MessageDefinitionInputParameter Properties**

The following are properties for MessageDefinitionInputParameter.

IN THIS SECTION:

booleanValue
A boolean input parameter.

booleanValues
A list of boolean parameters.

dateTimeValue
A datetime input parameter.
dateTimeValues
A list of datetime input parameters.

dateValue
A date input parameter.

dateValues
A list of date input parameters.

name
A name input parameter.

numberValue
A number input parameter.

numberValues
A list of number input parameters.

recordIdValue
A record ID input parameter.

recordIdValues
A list of record ID input parameters.

textValue
A text input parameter.

textValues
A list of text input parameters.

**booleanValue**
A boolean input parameter.

**Signature**

```csharp
public Boolean booleanValue {get; set;}
```

**Property Value**
Type: Boolean

**booleanValues**
A list of boolean parameters.

**Signature**

```csharp
public List<Boolean> booleanValues {get; set;}
```

**Property Value**
Type: List on page 3177<Boolean>
**dateTimeValue**
A datetime input parameter.

**Signature**
```java
public Datetime dateTimeValue {get; set;}
```

**Property Value**
Type: Datetime

**dateTimeValues**
A list of datetime input parameters.

**Signature**
```java
public List<Datetime> dateTimeValues {get; set;}
```

**Property Value**
Type: List on page 3177<Datetime>

**dateValue**
A date input parameter.

**Signature**
```java
public Date dateValue {get; set;}
```

**Property Value**
Type: Date

**dateValues**
A list of date input parameters.

**Signature**
```java
public List<Date> dateValues {get; set;}
```

**Property Value**
Type: List on page 3177<Date>

**name**
A name input parameter.
Signature
public String name {get; set;}

Property Value
Type: String

**numberValue**
A number input parameter.

Signature
public Double numberValue {get; set;}

Property Value
Type: Double

**numberValues**
A list of number input parameters.

Signature
public List<Double> numberValues {get; set;}

Property Value
Type: List on page 3177<Double>

**recordIdValue**
A record ID input parameter.

Signature
public String recordIdValue {get; set;}

Property Value
Type: String

**recordIdValues**
A list of record ID input parameters.

Signature
public List<String> recordIdValues {get; set;}
Property Value
Type: List on page 3177<String>

textValue
A text input parameter.

Signature
public String textValue {get; set;}

Property Value
Type: String
textValues
A list of text input parameters.

Signature
public List<String> textValues {get; set;}

Property Value
Type: List on page 3177<String>

PaymentItemStatus Enum
Represents the status of a payment item in payment requests sent in enhanced Messaging channels.

Enum Values
The following are the values of the RichMessaging.PaymentItemStatus enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FinalCost</td>
<td>Indicates that the payment item's cost is final and has been determined.</td>
</tr>
<tr>
<td>PendingCost</td>
<td>Indicates that the payment item's cost is pending and has not been determined yet.</td>
</tr>
</tbody>
</table>

PaymentLineItem Class
Represents a payment line item in payment requests sent in enhanced Messaging channels.

Namespace
RichMessaging
Example

```java
public with sharing class MessagingPaymentLineItems {

    @InvocableMethod
    public static List<List<RichMessaging.PaymentLineItem>> getLineItems() {
        Double amount = 0.25;
            RichMessaging.PaymentLineItem pizza = new RichMessaging.PaymentLineItem('pizza', amount);
            RichMessaging.PaymentLineItem pasta = new RichMessaging.PaymentLineItem('pasta', amount);
            pizza.statusValue = RichMessaging.PaymentItemStatus.FinalCost;
            pasta.statusValue = RichMessaging.PaymentItemStatus.FinalCost;

                pizza, pasta
            };
            result.add(options);
        return result;
    }
}
```

IN THIS SECTION:
- PaymentLineItem Constructors
- PaymentLineItem Properties
- PaymentLineItem Methods

PaymentLineItem Constructors

The following are constructors for PaymentLineItem.

IN THIS SECTION:
- PaymentLineItem(label, amount, timing)
  Creates a new instance of the RichMessaging.PaymentLineItem class.
- PaymentLineItem(label, amount)
  Creates a new instance of the RichMessaging.PaymentLineItem class.
- PaymentLineItem()
  Creates a new instance of the RichMessaging.PaymentLineItem class.

PaymentLineItem(label, amount, timing)

Creates a new instance of the RichMessaging.PaymentLineItem class.
Signature

public PaymentLineItem(String label, Double amount, RichMessaging.AbstractTiming timing)

Parameters

label
Type: String
The label of the payment line item.

amount
Type: Double
The amount of the payment line item.

timing
Type: RichMessaging.AbstractTiming
The timing of the payment line item.

PaymentLineItem(label, amount)
Creates a new instance of the RichMessaging.PaymentLineItem class.

Signature

public PaymentLineItem(String label, Double amount)

Parameters

label
Type: String
The label of the payment line item.

amount
Type: Double
The amount of the payment line item.

PaymentLineItem()
Creates a new instance of the RichMessaging.PaymentLineItem class.

Signature

public PaymentLineItem()

PaymentLineItem Properties
The following are properties for PaymentLineItem.
IN THIS SECTION:

- **amount**
  The amount of the payment line item.

- **amountValue**
  The amount value of the payment line item.

- **automaticReloadPaymentThresholdAmount**
  The automatic reload payment threshold amount of the payment line item.

- **automaticReloadPaymentThresholdAmountValue**
  The automatic reload payment threshold amount value of the payment line item.

- **label**
  The label of the payment line item.

- **labelValue**
  The label value of the payment line item.

- **lineItemType**
  The line item type of the payment line item. Read-only variable.

- **status**
  The status of the payment line item.

- **statusValue**
  The status value of the payment line item.

- **timing**
  The timing of the payment line item.

- **timingValue**
  The timing value of the payment line item.

**amount**

The amount of the payment line item.

**Signature**

```csharp
public Double amount {get; set;}
```

**Property Value**

Type: **Double**

**amountValue**

The amount value of the payment line item.

**Signature**

```csharp
public Double amountValue {get; set;}
```
Property Value
Type: Double

automaticReloadPaymentThresholdAmount
The automatic reload payment threshold amount of the payment line item.

Signature
public Double automaticReloadPaymentThresholdAmount {get; set;}

Property Value
Type: Double

automaticReloadPaymentThresholdAmountValue
The automatic reload payment threshold amount value of the payment line item.

Signature
public Double automaticReloadPaymentThresholdAmountValue {get; set;}

Property Value
Type: Double

label
The label of the payment line item.

Signature
public String label {get; set;}

Property Value
Type: String

labelValue
The label value of the payment line item.

Signature
public String labelValue {get; set;}

Property Value
Type: String
**lineItemType**
The line item type of the payment line item. Read-only variable.

**Signature**
public String lineItemType {get; set;}

**Property Value**
Type: String

**status**
The status of the payment line item.

**Signature**
public String status {get; set;}

**Property Value**
Type: String

**statusValue**
The status value of the payment line item.

**Signature**
public RichMessaging.PaymentItemStatus statusValue {get; set;}

**Property Value**
Type: RichMessaging.PaymentItemStatus

**timing**
The timing of the payment line item.

**Signature**
public RichMessaging.AbstractTiming timing {get; set;}

**Property Value**
Type: RichMessaging.AbstractTiming

**timingValue**
The timing value of the payment line item.
Signature

public RichMessaging.AbstractTiming timingValue {get; set;}

Property Value

Type: RichMessaging.AbstractTiming

PaymentLineItem Methods

The following are methods for PaymentLineItem.

PaymentMethod Class

Represents a payment method.

Namespace

RichMessaging

IN THIS SECTION:

PaymentMethod Constructors
PaymentMethod Properties

PaymentMethod Constructors

The following are constructors for PaymentMethod.

IN THIS SECTION:

PaymentMethod(network, paymentType, displayName)

PaymentMethod(network, paymentType, displayName)


Signature

public PaymentMethod(String network, String paymentType, String displayName)

Parameters

network
Type: String
The network associated with the payment method.

paymentType
Type: String
The payment type of the payment method.

**displayName**
Type: String
The display name of the payment method.

**PaymentMethod Properties**
The following are properties for **PaymentMethod**.

**IN THIS SECTION:**
- **displayName**
  The display name of the payment method.
- **network**
  The network associated with the payment method.
- **paymentType**
  The payment type of the payment method.

**displayName**
The display name of the payment method.

**Signature**
```java
public String displayName {get; set;}
```

**Property Value**
Type: String

**network**
The network associated with the payment method.

**Signature**
```java
public String network {get; set;}
```

**Property Value**
Type: String

**paymentType**
The payment type of the payment method.

**Signature**
```java
public String paymentType {get; set;}
```
PostalAddress Class

Represents the postal address.

Namespace

RichMessaging

PostalAddress Constructors

The following are constructors for PostalAddress.

PostalAddress(addressLines, subLocality, locality, postalCode, subAdministrativeArea, administrativeArea, country, countryCode)

Creates a new instance of the RichMessaging.PostalAddress class.

Signature

public PostalAddress(List<String> addressLines, String subLocality, String locality, String postalCode, String subAdministrativeArea, String administrativeArea, String country, String countryCode)

Parameters

addressLines
Type: List<String>
The street address.

subLocality
Type: String
The sub-locality of the address.

locality
Type: String
The locality of the address.
PostalAddress Properties

The following are properties for PostalAddress.

IN THIS SECTION:
  addressLines
  The street address.
  administrativeArea
  The administrative area.
  country
  The country.
  countryCode
  The country code.
  locality
  The locality of the address.
  postalCode
  The postal code.
  subAdministrativeArea
  The sub-administrative area.
  subLocality
  The sub-locality of the address.

addressLines
The street address.
Signature
public List<String> addressLines {get; set;}

Property Value
Type: List<String>

administrativeArea
The administrative area.

Signature
public String administrativeArea {get; set;}

Property Value
Type: String

country
The country.

Signature
public String country {get; set;}

Property Value
Type: String

countryCode
The country code.

Signature
public String countryCode {get; set;}

Property Value
Type: String

locality
The locality of the address.

Signature
public String locality {get; set;}

2680
Property Value
Type: String

**postalCode**
The postal code.

Signature
public String postalCode {get; set;}

Property Value
Type: String

**subAdministrativeArea**
The sub-administrative area.

Signature
public String subAdministrativeArea {get; set;}

Property Value
Type: String

**subLocality**
The sub-locality of the address.

Signature
public String subLocality {get; set;}

ProcessPaymentHandler Interface
Interface used to process payment requests.

Namespace
RichMessaging

IN THIS SECTION:
  ProcessPaymentHandler Methods
ProcessPaymentHandler Example Implementation

ProcessPaymentHandler Methods
The following are methods for ProcessPaymentHandler.

IN THIS SECTION:

processPaymentRequest(var1)
Processes a payment request.

processPaymentRequest(var1)
Processes a payment request.

Signature
public RichMessaging.ProcessPaymentResult
processPaymentRequest(RichMessaging.ProcessPaymentRequest var1)

Parameters
var1
Type: RichMessaging.ProcessPaymentRequest
The payment request.

Return Value
Type: RichMessaging.ProcessPaymentResult

ProcessPaymentRequest Class
Represents a request to process a payment.
Namespace

RichMessaging

IN THIS SECTION:

ProcessPaymentRequest Constructors
ProcessPaymentRequest Properties

ProcessPaymentRequest Constructors

The following are constructors for ProcessPaymentRequest.

IN THIS SECTION:

ProcessPaymentRequest(transactionIdentifier, paymentData, billingContact, shippingContact, paymentMethod, shippingMethod, contextRecordId)


ProcessPaymentRequest(transactionIdentifier, paymentData, billingContact, shippingContact, paymentMethod, shippingMethod, contextRecordId)


Signature


Parameters

transactionIdentifier
Type: String
The transaction identifier associated with the payment request.

paymentData
Type: String
The encrypted payment data for the payment request.

billingContact
Type: RichMessaging.AddressableContact
The billing contact information for the payment request.

shippingContact
Type: RichMessaging.AddressableContact
The shipping contact information for the payment request.

paymentMethod
Type: RichMessaging.PaymentMethod
The payment method for the payment request.

shippingMethod
Type: RichMessaging.ShippingMethod
The shipping method for the payment request.

contextRecordId
Type: String
The context record ID associated with the payment request.

ProcessPaymentRequest Properties
The following are properties for ProcessPaymentRequest.

IN THIS SECTION:

  billingContact
  The billing contact information for the payment request.

  contextRecordId
  The context record ID associated with the payment request.

  paymentData
  The encrypted payment data for the payment request.

  paymentMethod
  The payment method for the payment request.

  shippingContact
  The shipping contact information for the payment request.

  shippingMethod
  The shipping method for the payment request.

  transactionIdentifier
  The transaction identifier associated with the payment request.

billingContact
The billing contact information for the payment request.

Signature

public RichMessaging.AddressableContact billingContact {get; set;}

Property Value
Type: RichMessaging.AddressableContact

contextRecordId
The context record ID associated with the payment request.
Signature
public String contextRecordId {get; set;}

Property Value
Type: String

paymentData
The encrypted payment data for the payment request.

Signature
public String paymentData {get; set;}

Property Value
Type: String

paymentMethod
The payment method for the payment request.

Signature
public RichMessaging.PaymentMethod paymentMethod {get; set;}

Property Value
Type: RichMessaging.PaymentMethod

shippingContact
The shipping contact information for the payment request.

Signature
public RichMessaging.AddressableContact shippingContact {get; set;}

Property Value
Type: RichMessaging.AddressableContact

shippingMethod
The shipping method for the payment request.

Signature
public RichMessaging.ShippingMethod shippingMethod {get; set;}

2685
Property Value
Type: RichMessaging.ShippingMethod

**transactionIdentifier**
The transaction identifier associated with the payment request.

**Signature**
public String transactionIdentifier {get; set;}

Property Value
Type: String

## ProcessPaymentResult Class
Represents the result of a payment processing operation.

### Namespace
RichMessaging

### IN THIS SECTION:
- ProcessPaymentResult Constructors
- ProcessPaymentResult Properties

### ProcessPaymentResult Constructors
The following are constructors for ProcessPaymentResult.

**IN THIS SECTION:**
- ProcessPaymentResult(resultStatus, errorMessage)
- ProcessPaymentResult(resultStatus)

### ProcessPaymentResult (resultStatus, errorMessage)

**Signature**
public ProcessPaymentResult (RichMessaging.ProcessPaymentResultStatus resultStatus, String errorMessage)
Parameters

resultStatus
  Type: RichMessaging.ProcessPaymentResultStatus
  The status of the payment processing result.

errorMessage
  Type: String
  The error message associated with the payment processing result, if any.

ProcessPaymentResult (resultStatus)


Signature

public ProcessPaymentResult (RichMessaging.ProcessPaymentResultStatus resultStatus)

Parameters

resultStatus
  Type: RichMessaging.ProcessPaymentResultStatus
  The status of the payment processing result.

ProcessPaymentResult Properties

The following are properties for ProcessPaymentResult.

IN THIS SECTION:
  errorMessage
  The error message associated with the payment processing result, if any.
  resultStatus
  The status of the payment processing result.

errorMessage

The error message associated with the payment processing result, if any.

Signature

public String errorMessage {get; set;}

Property Value

Type: String

resultStatus

The status of the payment processing result.
Signature

```
public RichMessaging.ProcessPaymentResultStatus resultStatus {get; set;}
```

Property Value

Type: `RichMessaging.ProcessPaymentResultStatus`

**ProcessPaymentResultStatus Enum**

Represents the status of a payment processing result.

**Enum Values**

The following are the values of the `RichMessaging.ProcessPaymentResultStatus` enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PROCESSOR_ERROR</td>
<td>Indicates an error occurred during payment processing at the processor level.</td>
</tr>
<tr>
<td>SUCCESS</td>
<td>Indicates a successful payment processing result.</td>
</tr>
</tbody>
</table>

**RecurringTiming Class**

Represents a payment that occurs on a regular basis.

**Namespace**

`RichMessaging`

**IN THIS SECTION:**

- RecurringTiming Constructors
- RecurringTiming Properties

**RecurringTiming Constructors**

The following are constructors for `RecurringTiming`.

**IN THIS SECTION:**

- `RecurringTiming(startDate, endDate, intervalCount, intervalUnit)`
- `RecurringTiming()`

`RecurringTiming(startDate, endDate, intervalCount, intervalUnit)`

Signature

public RecurringTiming(Date startDate, Date endDate, Integer intervalCount, RichMessaging.TimingIntervalUnit intervalUnit)

Parameters

startDate
Type: Date
The start date. Invocable variable.

endDate
Type: Date
The end date. Invocable variable.

intervalCount
Type: Integer
The number of interval units that make up the total payment interval. Invocable variable.

intervalUnit
Type: RichMessaging.TimingIntervalUnit
The amount of time—in calendar units, such as day, month, or year—that represents a fraction of the total payment interval. Invocable variable.

RecurringTiming()


Signature

public RecurringTiming()

RecurringTiming Properties

The following are properties for RecurringTiming.

IN THIS SECTION:

endDate
The end date. Invocable variable.

endDateValue
The end date. Enabled for Lightning components.

intervalCount
The number of interval units that make up the total payment interval. Invocable variable.

intervalCountValue
The number of interval units that make up the total payment interval. Enabled for Lightning components.

intervalUnit
The amount of time—in calendar units, such as day, month, or year—that represents a fraction of the total payment interval. Invocable variable.
intervalUnitValue
The amount of time—in calendar units, such as day, month, or year—that represents a fraction of the total payment interval. Enabled for Lightning components.

startDate
The start date. Invocable variable.

startDateValue
The start date. Enabled for Lightning components.

timingType
Always returns “RecurringTiming”.

endDate
The end date. Invocable variable.

Signature
generic Date endDate {get; set;}

Property Value
Type: Date

endDateValue
The end date. Enabled for Lightning components.

Signature
generic Date endDateValue {get; set;}

Property Value
Type: Date

intervalCount
The number of interval units that make up the total payment interval. Invocable variable.

Signature
generic Integer intervalCount {get; set;}

Property Value
Type: Integer

intervalCountValue
The number of interval units that make up the total payment interval. Enabled for Lightning components.
**Signature**

```java
public Integer intervalCountValue {get; set;}
```

**Property Value**

Type: `Integer`

---

**intervalUnit**

The amount of time—in calendar units, such as day, month, or year—that represents a fraction of the total payment interval. Invocable variable.

**Signature**

```java
public String intervalUnit {get; set;}
```

**Property Value**

Type: `String`

---

**intervalUnitValue**

The amount of time—in calendar units, such as day, month, or year—that represents a fraction of the total payment interval. Enabled for Lightning components.

**Signature**

```java
public RichMessaging.TimingIntervalUnit intervalUnitValue {get; set;}
```

**Property Value**

Type: `RichMessaging.TimingIntervalUnit`

---

**startDate**

The start date. Invocable variable.

**Signature**

```java
public Date startDate {get; set;}
```

**Property Value**

Type: `Date`

---

**startDateValue**

The start date. Enabled for Lightning components.
Signature

```java
public Date startDateValue {get; set;}
```

Property Value

Type: Date

**timingType**

Always returns “RecurringTiming”.

Signature

```java
public String timingType {get; set;}
```

Property Value

Type: String

**ShippingMethod Class**

Represents a shipping method listed in payment requests sent in enhanced Messaging channels.

**Namespace**

RichMessaging

**Example**

```java
public with sharing class MessagingShippingMethods {
    @InvocableMethod
    public static List<List<RichMessaging.ShippingMethod>> getShippingMethods(){
        Double amount = 0.25;
            new RichMessaging.ShippingMethod('doordash', amount, '1 hour delivery to your door', 'ddash'),
            new RichMessaging.ShippingMethod('UPS', amount, '2 days delivery', 'UPS')
        };
        result.add(options);
        return result;
    }
}
```
ShippingMethod Constructors

The following are constructors for ShippingMethod.

ShippingMethod(label, amount, detail, identifier)


Signature

public ShippingMethod(String label, Double amount, String detail, String identifier)

Parameters

label
Type: String
The label of the shipping method.

amount
Type: Double
The amount of the shipping method.

detail
Type: String
Details about the shipping method.

identifier
Type: String
The identifier of the shipping method.

ShippingMethod()


Signature

public ShippingMethod()
ShippingMethod Properties
The following are properties for ShippingMethod.

IN THIS SECTION:
  amount
  The amount of the shipping method.
  amountValue
  The amount value of the shipping method.
  detail
  Details about the shipping method.
  detailValue
  The detail value of the shipping method.
  identifier
  The identifier of the shipping method.
  identifierValue
  The identifier value of the shipping method.
  label
  The label of the shipping method.
  labelValue
  The label value of the shipping method.
  shippingMethodType
  The shipping method type. Read only.

amount
The amount of the shipping method.

Signature
public Double amount {get; set;}

Property Value
Type: Double

amountValue
The amount value of the shipping method.

Signature
public Double amountValue {get; set;}
Property Value
Type: Double

detail
Details about the shipping method.

Signature
public String detail {get; set;}

Property Value
Type: String

detailValue
The detail value of the shipping method.

Signature
public String detailValue {get; set;}

Property Value
Type: String

identifier
The identifier of the shipping method.

Signature
public String identifier {get; set;}

Property Value
Type: String

identifierValue
The identifier value of the shipping method.

Signature
public String identifierValue {get; set;}

Property Value
Type: String
**label**
The label of the shipping method.

**Signature**
public String label {get; set;}

**Property Value**
Type: String

**labelValue**
The label value of the shipping method.

**Signature**
public String labelValue {get; set;}

**Property Value**
Type: String

**shippingMethodType**
The shipping method type. Read only.

**Signature**
public String shippingMethodType {get; set;}

**Property Value**
Type: String

**TimeSlotOption Class**
Represents a complex time slot option type. This class is used to provide time option payloads that can be translated to structured content payloads in rich content messages.

**Namespace**
RichMessaging

IN THIS SECTION:
  - TimeSlotOption Constructors
  - TimeSlotOption Properties
TimeSlotOption Constructors

The following are constructors for TimeSlotOption.

IN THIS SECTION:

TimeSlotOption(startTime, endTime)
Creates a TimeSlotOption object with a start and end time.

TimeSlotOption(startTime, duration)
Creates a TimeSlotOption object with a start time and a duration.

TimeSlotOption()
Creates a TimeSlotOption object.

TimeSlotOption(startTime, endTime)

Creates a TimeSlotOption object with a start and end time.

Signature

public TimeSlotOption(Datetime startTime, Datetime endTime)

Parameters

startTime
Type: Datetime
Start time.

endTime
Type: Datetime
End time.

TimeSlotOption(startTime, duration)

Creates a TimeSlotOption object with a start time and a duration.

Signature

public TimeSlotOption(Datetime startTime, Integer duration)

Parameters

startTime
Type: Datetime
Start time.

duration
Type: Integer
Duration in seconds.
**TimeSlotOption()**

Creates a `TimeSlotOption` object.

**Signature**

```java
public TimeSlotOption()
```

**TimeSlotOption Properties**

The following are properties for `TimeSlotOption`.

**IN THIS SECTION:**

- `duration`
  - The duration in seconds.

- `durationValue`
  - The duration in seconds. Enabled for Lightning components.

- `endTimeValue`
  - The end time. Enabled for Lightning components.

- `startTime`
  - The start time.

- `startTimeValue`
  - The start time. Enabled for Lightning components.

**duration**

The duration in seconds.

**Signature**

```java
public Integer duration {get; set;}
```

**Property Value**

Type: `Integer`

**durationValue**

The duration in seconds. Enabled for Lightning components.

**Signature**

```java
public Integer durationValue {get; set;}
```

**Property Value**

Type: `Integer`
**endTimeValue**
The end time. Enabled for Lightning components.

**Signature**
public Datetime endTimeValue {get; set;}

**Property Value**
Type: Datetime

**startTime**
The start time.

**Signature**
public Datetime startTime {get; set;}

**Property Value**
Type: Datetime

**startTimeValue**
The start time. Enabled for Lightning components.

**Signature**
public Datetime startTimeValue {get; set;}

**Property Value**
Type: Datetime

## TimingIntervalUnit Enum
Represents an enumerated type that describes the timing interval.

### Enum Values
The following are the values of the RichMessaging.TimingIntervalUnit enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Day</td>
<td>Day interval.</td>
</tr>
<tr>
<td>Hour</td>
<td>Hour interval.</td>
</tr>
<tr>
<td>Minute</td>
<td>Minute interval.</td>
</tr>
</tbody>
</table>
TimingType Enum

Represents an enumerated type that describes the type of timing.

Enum Values

The following are the values of the RichMessaging.TimingType enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DeferredTiming</td>
<td>Indicates that the timing is deferred. See DeferredTiming Class.</td>
</tr>
<tr>
<td>RecurringTiming</td>
<td>Indicates that the timing recurs. See RecurringTiming Class.</td>
</tr>
</tbody>
</table>

Schema Namespace

The Schema namespace provides classes and methods for schema metadata information.

The following are the classes in the Schema namespace.

IN THIS SECTION:

- ChildRelationship Class
  Contains methods for accessing the child relationship as well as the child sObject for a parent sObject.

- DataCategory Class
  Represents the categories within a category group.

- DataCategoryGroupSObjectTypePair Class
  Specifies a category group and an associated object.

- DescribeColorResult Class
  Contains color metadata information for a tab.

- DescribeDataCategoryGroupResult Class
  Contains the list of the category groups associated with KnowledgeArticleVersion and Question.

- DescribeDataCategoryGroupStructureResult Class
  Contains the category groups and categories associated with KnowledgeArticleVersion and Question.

- DescribeFieldResult Class
  Contains methods for describing sObject fields.

- DescribeIconResult Class
  Contains icon metadata information for a tab.
DescribeSObjectResult Class
Contains methods for describing SObjects. None of the methods take an argument.

DescribeTabResult Class
Contains tab metadata information for a tab in a standard or custom app available in the Salesforce user interface.

DescribeTabSetResult Class
Contains metadata information about a Salesforce Classic standard or custom app available in the Salesforce user interface.

DisplayType Enum
A Schema.DisplayType enum value is returned by the field describe result's getType method.

FieldDescribeOptions Enum
A Schema.FieldDescribeOptions enum value is a parameter in the SObjectType.getDescribe method.

FieldSet Class
Contains methods for discovering and retrieving the details of field sets created on sObjects.

FieldSetMember Class
Contains methods for accessing the metadata for field set member fields.

PicklistEntry Class
Represents a picklist entry.

RecordTypeInfo Class
Contains methods for accessing record type information for an sObject with associated record types.

SOAPType Enum
A Schema.SOAPType enum value is returned by the field describe result getSoapType method.

SObjectDescribeOptions Enum
A Schema.SObjectDescribeOptions enum value is a parameter in the SObjectType.getDescribe method.

SObjectField Class
A Schema.sObjectField object is returned from the field describe result using the getController and getSObjectField methods.

SObjectType Class
A Schema.sObjectType object is returned from the field describe result using the getReferenceTo method, or from the sObject describe result using the getSObjectType method.

ChildRelationship Class
Contains methods for accessing the child relationship as well as the child sObject for a parent sObject.

Namespace
Schema

Example
A ChildRelationship object is returned from the sObject describe result using the getChildRelationship method. For example:

```java
Schema.DescribeSObjectResult R = Account.SObjectType.getDescribe();
```
**ChildRelationship Methods**

The following are methods for `ChildRelationship`. All are instance methods.

**IN THIS SECTION:**

- `getChildSObject()`: Returns the token of the child sObject on which there is a foreign key back to the parent sObject.
- `getField()`: Returns the token of the field that has a foreign key back to the parent sObject.
- `getRelationshipName()`: Returns the name of the relationship.
- `isCascadeDelete()`: Returns `true` if the child object is deleted when the parent object is deleted, `false` otherwise.
- `is DeprecatedAndHidden()`: Reserved for future use.
- `isRestrictedDelete()`: Returns `true` if the parent object can't be deleted because it is referenced by a child object, `false` otherwise.

### `getChildSObject()`

Returns the token of the child sObject on which there is a foreign key back to the parent sObject.

**Signature**

```
public Schema.SObjectType getChildSObject()
```

**Return Value**

Type: `Schema.SObjectType`

### `getField()`

Returns the token of the field that has a foreign key back to the parent sObject.

**Signature**

```
public Schema.SObjectField getField()
```

**Return Value**

Type: `Schema.SObjectField`

### `getRelationshipName()`

Returns the name of the relationship.
**Signature**

```java
public String getRelationshipName()
```

**Return Value**

Type: String

**isCascadeDelete()**

Returns `true` if the child object is deleted when the parent object is deleted, `false` otherwise.

**Signature**

```java
public Boolean isCascadeDelete()
```

**Return Value**

Type: Boolean

**isDeprecatedAndHidden()**

Reserved for future use.

**Signature**

```java
public Boolean isDeprecatedAndHidden()
```

**Return Value**

Type: Boolean

**isRestrictedDelete()**

Returns `true` if the parent object can’t be deleted because it is referenced by a child object, `false` otherwise.

**Signature**

```java
public Boolean isRestrictedDelete()
```

**Return Value**

Type: Boolean

---

**DataCategory Class**

Represents the categories within a category group.

**Namespace**

Schema
Usage

The Schema.DataCategory object is returned by the getTopCategories method.

DataCategory Methods

The following are methods for DataCategory. All are instance methods.

IN THIS SECTION:

getChildCategories()
Returns a recursive object that contains the visible sub categories in the data category.

getLabel()
Returns the label for the data category used in the Salesforce user interface.

getName()
Returns the unique name used by the API to access to the data category.

getChildCategories()

Returns a recursive object that contains the visible sub categories in the data category.

Signature

public Schema.DataCategory getChildCategories()

Return Value

Type: List<Schema.DataCategory>

getLabel()

Returns the label for the data category used in the Salesforce user interface.

Signature

public String getLabel()

Return Value

Type: String

getName()

Returns the unique name used by the API to access to the data category.

Signature

public String getName()
Return Value
Type: String

DataCategoryGroupSobjectTypePair Class
Specifies a category group and an associated object.

Namespace
Schema

Usage
Schema.DataCategoryGroupSobjectTypePair is used by the describeDataCategory GroupStructures method to return the categories available to this object.

IN THIS SECTION:
DataCategoryGroupSobjectTypePair Constructors
DataCategoryGroupSobjectTypePair Methods

DataCategoryGroupSobjectTypePair Constructors
The following are constructors for DataCategoryGroupSobjectTypePair.

IN THIS SECTION:
DataCategoryGroupSobjectTypePair()
Creates a new instance of the Schema.DataCategoryGroupSobjectTypePair class.

DataCategoryGroupSobjectTypePair()
Creates a new instance of the Schema.DataCategoryGroupSobjectTypePair class.

Signature
public DataCategoryGroupSobjectTypePair()

DataCategoryGroupSobjectTypePair Methods
The following are methods for DataCategoryGroupSobjectTypePair. All are instance methods.

IN THIS SECTION:
getDataCategoryGroupName() Returns the unique name used by the API to access the data category group.
getSobject() Returns the object name associated with the data category group.
setDataCategoryGroupName(name)
Specifies the unique name used by the API to access the data category group.

setSobject(sObjectName)
Sets the sObject associated with the data category group.

getDataCategoryGroupName()
Returns the unique name used by the API to access the data category group.

Signature
public String getDataCategoryGroupName()

Return Value
Type: String

getSobject()
Returns the object name associated with the data category group.

Signature
public String getSobject()

Return Value
Type: String

setDataCategoryGroupName(name)
Specifies the unique name used by the API to access the data category group.

Signature
public String setDataCategoryGroupName(String name)

Parameters
name
Type: String

Return Value
Type: Void

setSobject(sObjectName)
Sets the sObject associated with the data category group.
Signature

public Void setSobject(String sObjectName)

Parameters

sObjectName
Type: String

The sObjectName is the object name associated with the data category group. Valid values are:

- KnowledgeArticleVersion—for article types.
- Question—for questions from Answers.

Return Value
Type: Void

DescribeColorResult Class

Contains color metadata information for a tab.

Namespace

Schema

Usage

The getColors method of the Schema.DescribeTabResult class returns a list of Schema.DescribeColorResult objects that describe colors used in a tab.

The methods in the Schema.DescribeColorResult class can be called using their property counterparts. For each method starting with get, you can omit the get prefix and the ending parentheses () to call the property counterpart. For example, colorResultObj.color is equivalent to colorResultObj.getColor().

Example

This sample shows how to get the color information in the Sales app for the first tab's first color.

```apex
// Get tab set describes for each app
List<Schema.DescribeTabSetResult> tabSetDesc = Schema.DescribeTabs();

// Iterate through each tab set describe for each app and display the info
for (Schema.DescribeTabSetResult tsr : tabSetDesc) {
    // Display tab info for the Sales app
    if (tsr.getLabel() == 'Sales') {
        // Get color information for the first tab
        List<Schema.DescribeColorResult> colorDesc = tsr.getTabs()[0].getColors();
        // Display the icon color, theme, and context of the first color returned
        System.debug('Color: ' + colorDesc[0].getColor());
        System.debug('Theme: ' + colorDesc[0].getTheme());
        System.debug('Context: ' + colorDesc[0].getContext());
    }
}
DescribeColorResult Methods

The following are methods for DescribeColorResult. All are instance methods.

IN THIS SECTION:

getColor()
Returns the Web RGB color code, such as 00FF00.

getContext()
Returns the color context. The context determines whether the color is the main color for the tab or not.

getTheme()
Returns the color theme.

gColor()
Returns the Web RGB color code, such as 00FF00.

Signature

public String getColor()

Return Value
Type: String

gContext()
Returns the color context. The context determines whether the color is the main color for the tab or not.

Signature

public String getContext()

Return Value
Type: String

gTheme()
Returns the color theme.
public String getTheme()

Return Value
Type: String
Possible theme values include theme3, theme4, and custom.

- theme3 is the Salesforce theme introduced during Spring ’10.
- theme4 is the Salesforce theme introduced in Winter ’14 for the mobile touchscreen version of Salesforce.
- custom is the theme name associated with a custom icon.

DescribeDataCategoryGroupResult Class
Contains the list of the category groups associated with KnowledgeArticleVersion and Question.

Namespace
Schema

Usage
The describeDataCategoryGroups method returns a Schema.DescribeDataCategoryGroupResult object containing the list of the category groups associated with the specified object.

For additional information and code examples using describeDataCategoryGroups, see Accessing All Data Categories Associated with an sObject.

Example
The following is an example of how to instantiate a data category group describe result object:

```java
List<String> objType = new List<String>();
objType.add('KnowledgeArticleVersion');
objType.add('Question');

List<Schema.DescribeDataCategoryGroupResult> describeCategoryResult =
    Schema.describeDataCategoryGroups(objType);
```

DescribeDataCategoryGroupResult Methods
The following are methods for DescribeDataCategoryGroupResult. All are instance methods.

IN THIS SECTION:
- getCategoryCount()
  Returns the number of visible data categories in the data category group.
- getDescription()
  Returns the description of the data category group.
getLabel()
Returns the label for the data category group used in the Salesforce user interface.

getName()
Returns the unique name used by the API to access to the data category group.

getSobject()
Returns the object name associated with the data category group.

getCategoryCount()
Returns the number of visible data categories in the data category group.

**Signature**

```java
public Integer getCategoryCount()
```

**Return Value**

Type: Integer

ggetDescription()
Returns the description of the data category group.

**Signature**

```java
public String getDescription()
```

**Return Value**

Type: String

getLabel()
Returns the label for the data category group used in the Salesforce user interface.

**Signature**

```java
public String getLabel()
```

**Return Value**

Type: String

getName()
Returns the unique name used by the API to access to the data category group.

**Signature**

```java
public String getName()
```
Return Value
Type: String

getSobject()
Returns the object name associated with the data category group.

Signature
public String getSobject()

Return Value
Type: String

DescribeDataCategoryGroupStructureResult Class
Contains the category groups and categories associated with KnowledgeArticleVersion and Question.

Namespace
Schema

Usage
The describeDataCategoryGroupStructures method returns a list of Schema.DescribeDataCategoryGroupStructureResult objects containing the category groups and categories associated with the specified object. For additional information and code examples, see Accessing All Data Categories Associated with an sObject.

Example
The following is an example of how to instantiate a data category group structure describe result object:

```java
List<DataCategoryGroupSobjectTypePair> pairs =
    new List<DataCategoryGroupSobjectTypePair>();

DataCategoryGroupSobjectTypePair pair1 =
    new DataCategoryGroupSobjectTypePair();
pair1.setSobject('KnowledgeArticleVersion');
pair1.setDataCategoryGroupName('Regions');

DataCategoryGroupSobjectTypePair pair2 =
    new DataCategoryGroupSobjectTypePair();
pair2.setSobject('Questions');
pair2.setDataCategoryGroupName('Regions');

pairs.add(pair1);
pairs.add(pair2);
```
DescribeDataCategoryGroupStructureResult Methods

The following are methods for DescribeDataCategoryGroupStructureResult. All are instance methods.

IN THIS SECTION:
- getDescription()
- getLabel()
- getName()
- getSobject()
- getTopCategories()

getDescription()

Returns the description of the data category group.

Signature

public String getDescription()

Return Value

Type: String

getLabel()

Returns the label for the data category group used in the Salesforce user interface.

Signature

public String getLabel()

Return Value

Type: String

getName()

Returns the unique name used by the API to access to the data category group.

Signature

public String getName()

Return Value

Type: String

getTopCategories()

Returns a Schema.DataCategory object, that contains the top categories visible depending on the user's data category group visibility settings.

Signature

public List<Schema.DataCategory> getTopCategories()

Return Value

Type: List<Schema.DataCategory>
**Signature**

```java
public String getName()
```

**Return Value**

Type: String

**getSobject()**

Returns the name of object associated with the data category group.

**Signature**

```java
public String getSobject()
```

**Return Value**

Type: String

**getTopCategories()**

Returns a `Schema.DataCategory` object, that contains the top categories visible depending on the user's data category group visibility settings.

**Signature**

```java
public List<Schema.DataCategory> getTopCategories()
```

**Return Value**

Type: `List<Schema.DataCategory>`

**Usage**

For more information on data category group visibility, see “Data Category Visibility” in the Salesforce online help.

**DescribeFieldResult Class**

Contains methods for describing sObject fields.

**Namespace**

`Schema`

**Usage**

Instances of field describe results on the same `DescribeFieldResult` aren't guaranteed to be equal because the state and behavior of a describe object is determined by various factors including the API version used. To compare describe results, call the
getSObjectField() method on the field describe results and use the equality operator (==) to compare the SObjectField values.

Example

The following is an example of how to instantiate a field describe result object:

```java
Schema.DescribeFieldResult dfr = Account.Description.getDescribe();
```

DescribeFieldResult Methods

The following are methods for DescribeFieldResult. All are instance methods.

IN THIS SECTION:

- `getByteLength()`: For variable-length fields (including binary fields), returns the maximum size of the field, in bytes.
- `getCalculatedFormula()`: Returns the formula specified for this field.
- `getController()`: Returns the token of the controlling field.
- `getDefaultValue()`: Returns the default value for this field.
- `getDefaultValueFormula()`: Returns the default value specified for this field if a formula is not used.
- `getDigits()`: Returns the maximum number of digits specified for the field. This method is only valid with Integer fields.
- `getInlineHelpText()`: Returns the content of the field-level help.
- `getLabel()`: Returns the text label that is displayed next to the field in the Salesforce user interface. This label can be localized.
- `getLength()`: Returns the maximum size of the field for the DescribeFieldResult object in Unicode characters (not bytes).
- `getLocalName()`: Returns the name of the field, similar to the `getName` method. However, if the field is part of the current namespace, the namespace portion of the name is omitted.
- `getName()`: Returns the field name used in Apex.
- `getPicklistValues()`: Returns a list of PicklistEntry objects. A runtime error is returned if the field is not a picklist.
- `getPrecision()`: For fields of type Double, returns the maximum number of digits that can be stored, including all numbers to the left and to the right of the decimal point (but excluding the decimal point character).
getReferenceTargetField()
Returns the name of the custom field on the parent standard or custom object whose values are matched against the values of the child external object’s indirect lookup relationship field. The match is done to determine which records are related to each other.

gerReferenceTo()
Returns a list of Schema.sObjectType objects for the parent objects of this field. If the isNamePointing method returns true, there is more than one entry in the list, otherwise there is only one.

gerRelationshipName()
Returns the name of the relationship.

gerRelationshipOrder()
Returns 1 if the field is a child, 0 otherwise.

gerScale()
For fields of type Double, returns the number of digits to the right of the decimal point. Any extra digits to the right of the decimal point are truncated.

gerSOAPType()
Returns one of the SoapType enum values, depending on the type of field.

gerSObjectField()
Returns the token for this field.

gerSObjectType()
Returns the Salesforce object type from which this field originates.

gerType()
Returns one of the DisplayType enum values, depending on the type of field.

isAccessible()
Returns true if the current user can see this field, false otherwise.

isAiPredictionField() (Beta)
Returns true if the current field is enabled to display Einstein prediction data, false otherwise.

isAutoNumber()
Returns true if the field is an Auto Number field, false otherwise.

isCalculated()
Returns true if the field is a custom formula field, false otherwise. Note that custom formula fields are always read-only.

isCascadeDelete()
Returns true if the child object is deleted when the parent object is deleted, false otherwise.

isCaseSensitive()
Returns true if the field is case sensitive, false otherwise.

isCreateable()
Returns true if the field can be created by the current user, false otherwise.

isCustom()
Returns true if the field is a custom field, false if it is a standard field, such as Name.

isDefaultedOnCreate()
Returns true if the field receives a default value when created, false otherwise.

isDependentPicklist()
Returns true if the picklist is a dependent picklist, false otherwise.
isDeprecatedAndHidden()
Reserved for future use.

isEncrypted()
Returns true if the field is encrypted with Shield Platform Encryption, false otherwise.

isExternalId()
Returns true if the field is used as an external ID, false otherwise.

isFilterable()
Returns true if the field can be used as part of the filter criteria of a WHERE statement, false otherwise.

isFormulaTreatNullNumberAsZero()
Returns true if null is treated as zero in a formula field, false otherwise.

isGroupable()
Returns true if the field can be included in the GROUP BY clause of a SOQL query, false otherwise. This method is only available for Apex classes and triggers saved using API version 18.0 and higher.

isHtmlFormatted()
Returns true if the field has been formatted for HTML and should be encoded for display in HTML, false otherwise. One example of a field that returns true for this method is a hyperlink custom formula field. Another example is a custom formula field that has an IMAGE text function.

isIdLookup()
Returns true if the field can be used to specify a record in an upsert method, false otherwise.

isNameField()
Returns true if the field is a name field, false otherwise.

isNamePointing()
Returns true if the field can have multiple types of objects as parents. For example, a task can have both the Contact/Lead ID (WhoId) field and the Opportunity/Account ID (WhatId) field return true for this method, because either of those objects can be the parent of a particular task record. This method returns false otherwise.

isNillable()
Returns true if the field is nillable, false otherwise. A nillable field can have empty content. A non-nillable field must have a value for the object to be created or saved.

isPermissionable()
Returns true if field permissions can be specified for the field, false otherwise.

isRestrictedDelete()
Returns true if the parent object can't be deleted because it is referenced by a child object, false otherwise.

isRestrictedPicklist()
Returns true if the field is a restricted picklist, false otherwise.

isSearchPrefilterable()
Returns true if a foreign key can be included in prefiltering when used in a SOSL WHERE clause, false otherwise.

isSortable()
Returns true if a query can sort on the field, false otherwise.

isUnique()
Returns true if the value for the field must be unique, false otherwise.
isUpdateable()  
Returns true if the field can be edited by the current user, or child records in a master-detail relationship field on a custom object can be reparented to different parent records; false otherwise.

isWriteRequiresMasterRead()  
Returns true if writing to the detail object requires read sharing instead of read/write sharing of the parent.

getByteLength()  
For variable-length fields (including binary fields), returns the maximum size of the field, in bytes.

Signature  
public Integer getByteLength()

Return Value  
Type: Integer

getCalculatedFormula()  
Returns the formula specified for this field.

Signature  
public String getCalculatedFormula()

Return Value  
Type: String

getController()  
Returns the token of the controlling field.

Signature  
public Schema.sObjectField getController()

Return Value  
Type: Schema.SObjectField

getDefaultValue()  
Returns the default value for this field.

Signature  
public Object getDefaultValue()
Return Value
Type: Object

getDefaultValueFormula()  
Returns the default value specified for this field if a formula is not used.

Signature
public String getDefaultValueFormula()

Return Value
Type: String

getDigits()  
Returns the maximum number of digits specified for the field. This method is only valid with Integer fields.

Signature
public Integer getDigits()

Return Value
Type: Integer

getInlineHelpText()  
Returns the content of the field-level help.

Signature
public String getInlineHelpText()

Return Value
Type: String

Usage
For more information, see “Define Field-Level Help” in the Salesforce online help.

getLabel()  
Returns the text label that is displayed next to the field in the Salesforce user interface. This label can be localized.

Signature
public String getLabel()
Return Value
Type: String

Usage

Note: For the Type field on standard objects, getLabel returns a label different from the default label. It returns a label of the form Object Type, where Object is the standard object label. For example, for the Type field on Account, getLabel returns Account Type instead of the default label Type. If the Type label is renamed, getLabel returns the new label. You can check or change the labels of all standard object fields from Setup by entering Rename Tabs and Labels in the Quick Find box, then selecting Rename Tabs and Labels.

**getLength()**
Returns the maximum size of the field for the DescribeFieldResult object in Unicode characters (not bytes).

Signature

```java
public Integer getLength()
```

**Return Value**
Type: Integer

**getLocalName()**
Returns the name of the field, similar to the getName method. However, if the field is part of the current namespace, the namespace portion of the name is omitted.

Signature

```java
public String getLocalName()
```

**Return Value**
Type: String

**getName()**
Returns the field name used in Apex.

Signature

```java
public String getName()
```

**Return Value**
Type: String
getPicklistValues()
Returns a list of PicklistEntry objects. A runtime error is returned if the field is not a picklist.

Signature
public List<Schema.PicklistEntry> getPicklistValues()

Return Value
Type: List<Schema.PicklistEntry>

getPrecision()
For fields of type Double, returns the maximum number of digits that can be stored, including all numbers to the left and to the right of the decimal point (but excluding the decimal point character).

Signature
public Integer getPrecision()

Return Value
Type: Integer

getReferenceTargetField()
Returns the name of the custom field on the parent standard or custom object whose values are matched against the values of the child external object’s indirect lookup relationship field. The match is done to determine which records are related to each other.

Signature
public String getReferenceTargetField()

Return Value
Type: String

Usage
For information about indirect lookup relationships, see “Indirect Lookup Relationship Fields on External Objects” in the Salesforce Help.

getReferenceTo()
Returns a list of Schema.sObjectType objects for the parent objects of this field. If the isNamePointing method returns true, there is more than one entry in the list, otherwise there is only one.

Signature
public List<Schema.sObjectType> getReferenceTo()
Return Value
Type: List<Schema.sObjectType>

Versioned Behavior Changes
In API version 51.0 and later, the getReferenceTo() method returns referenced objects that aren’t accessible to the context user. If the context user has access to an object’s field that references another object, irrespective of the context user’s access to the cross-referenced object, the method returns references. In API version 50.0 and earlier, if the context user doesn’t have access to the cross-referenced object, the method returns an empty list.

getRelationshipName()
Returns the name of the relationship.

Signature
public String getRelationshipName()

Return Value
Type: String

Usage
For more information about relationships and relationship names, see Understanding Relationship Names in the SOQL and SOSL Reference.

getRelationshipOrder()
Returns 1 if the field is a child, 0 otherwise.

Signature
public Integer getRelationshipOrder()

Return Value
Type: Integer

Usage
For more information about relationships and relationship names, see Understanding Relationship Names in the SOQL and SOSL Reference.

getScale()
For fields of type Double, returns the number of digits to the right of the decimal point. Any extra digits to the right of the decimal point are truncated.

Signature
public Integer getScale()
Return Value
Type: Integer

Usage
This method returns a fault response if the number has too many digits to the left of the decimal point.

getSOAPType()
Returns one of the SoapType enum values, depending on the type of field.

Signature
public Schema.SOAPType getSOAPType()

Return Value
Type: Schema.SOAPType

g getSObjectField()
Returns the token for this field.

Signature
public Schema.sObjectField getSObjectField()

Return Value
Type: Schema.sObjectField

g getSObjectType()
Returns the Salesforce object type from which this field originates.

Signature
public Schema.SObjectType getSObjectType()

Return Value
Type: Schema.SObjectType

Example
```
Schema.DescribeFieldResult f = Account.Industry.getDescribe();
Schema.sObjectType sourceType = f.getSObjectType();
Assert.areEqual(Account.sObjectType, sourceType);
```
**getType()**

Returns one of the DisplayType enum values, depending on the type of field.

**Signature**

`public Schema.DisplayType getType()`

**Return Value**

Type: `Schema.DisplayType`

**isAccessible()**

Returns `true` if the current user can see this field, `false` otherwise.

**Signature**

`public Boolean isAccessible()`

**Return Value**

Type: `Boolean`

**isAiPredictionField() (Beta)**

Returns `true` if the current field is enabled to display Einstein prediction data, `false` otherwise.

**Signature**

`public Boolean isAiPredictionField()`

**Return Value**

Type: `Boolean`

**Usage**

Custom number fields can be set to display Einstein prediction values. If you are participating in the Einstein Prediction Builder Beta program, use Einstein Prediction Builder to set up the value to display. Use this method to find out if a field is enabled to display an Einstein prediction value.

**isAutoNumber()**

Returns `true` if the field is an Auto Number field, `false` otherwise.

**Signature**

`public Boolean isAutoNumber()`
Return Value
Type: Boolean

Usage
Analogous to a SQL IDENTITY type, Auto Number fields are read-only, non-createable text fields with a maximum length of 30 characters. Auto Number fields are used to provide a unique ID that is independent of the internal object ID (such as a purchase order number or invoice number). Auto Number fields are configured entirely in the Salesforce user interface.

isCalculated()
Returns true if the field is a custom formula field, false otherwise. Note that custom formula fields are always read-only.

Signature
public Boolean isCalculated()

Return Value
Type: Boolean

isCascadeDelete()
Returns true if the child object is deleted when the parent object is deleted, false otherwise.

Signature
public Boolean isCascadeDelete()

Return Value
Type: Boolean

isCaseSensitive()
Returns true if the field is case sensitive, false otherwise.

Signature
public Boolean isCaseSensitive()

Return Value
Type: Boolean

isCreateable()
Returns true if the field can be created by the current user, false otherwise.
Signature

public Boolean isCreateable()

Return Value

Type: Boolean

isCustom()

Returns true if the field is a custom field, false if it is a standard field, such as Name.

Signature

public Boolean isCustom()

Return Value

Type: Boolean

isDefaultedOnCreate()

Returns true if the field receives a default value when created, false otherwise.

Signature

public Boolean isDefaultedOnCreate()

Return Value

Type: Boolean

Usage

If this method returns true, Salesforce implicitly assigns a value for this field when the object is created, even if a value for this field is not passed in on the create call. For example, in the Opportunity object, the Probability field has this attribute because its value is derived from the Stage field. Similarly, the Owner has this attribute on most objects because its value is derived from the current user (if the Owner field is not specified).

isDependentPicklist()

Returns true if the picklist is a dependent picklist, false otherwise.

Signature

public Boolean isDependentPicklist()

Return Value

Type: Boolean
isDeprecatedAndHidden()
Reserved for future use.

Signature
public Boolean isDeprecatedAndHidden()

Return Value
Type: Boolean

isEncrypted()
Returns true if the field is encrypted with Shield Platform Encryption, false otherwise.

Signature
public Boolean isEncrypted()

Return Value
Type: Boolean

isExternalID()
Returns true if the field is used as an external ID, false otherwise.

Signature
public Boolean isExternalID()

Return Value
Type: Boolean

isFilterable()
Returns true if the field can be used as part of the filter criteria of a WHERE statement, false otherwise.

Signature
public Boolean isFilterable()

Return Value
Type: Boolean

isFormulaTreatNullNumberAsZero()
Returns true if null is treated as zero in a formula field, false otherwise.
Signature
public Boolean isFormulaTreatNullNumberAsZero()

Return Value
Type: Boolean

isGroupable()
Returns true if the field can be included in the GROUP BY clause of a SOQL query, false otherwise. This method is only available for Apex classes and triggers saved using API version 18.0 and higher.

Signature
public Boolean isGroupable()

Return Value
Type: Boolean

isHtmlFormatted()
Returns true if the field has been formatted for HTML and should be encoded for display in HTML, false otherwise. One example of a field that returns true for this method is a hyperlink custom formula field. Another example is a custom formula field that has an IMAGE text function.

Signature
public Boolean isHtmlFormatted()

Return Value
Type: Boolean

isIdLookup()
Returns true if the field can be used to specify a record in an upsert method, false otherwise.

Signature
public Boolean isIdLookup()

Return Value
Type: Boolean

isNameField()
Returns true if the field is a name field, false otherwise.
Signature

public Boolean isNameField()

Return Value
Type: Boolean

Usage
This method is used to identify the name field for standard objects (such as AccountName for an Account object) and custom objects. Objects can only have one name field, except where the FirstName and LastName fields are used instead (such as on the Contact object). If a compound name is present, for example, the Name field on a person account, isNameField is set to true for that record.

isNamePointing()

Returns true if the field can have multiple types of objects as parents. For example, a task can have both the Contact/Lead ID (WhoId) field and the Opportunity/Account ID (WhatId) field return true for this method, because either of those objects can be the parent of a particular task record. This method returns false otherwise.

Signature

public Boolean isNamePointing()

Return Value
Type: Boolean

isNillable()

Returns true if the field is nillable, false otherwise. A nillable field can have empty content. A non-nillable field must have a value for the object to be created or saved.

Signature

public Boolean isNillable()

Return Value
Type: Boolean

isPermissionable()

Returns true if field permissions can be specified for the field, false otherwise.

Signature

public Boolean isPermissionable()
Return Value
Type: Boolean

**isRestrictedDelete()**
Returns `true` if the parent object can’t be deleted because it is referenced by a child object, `false` otherwise.

**Signature**
```java
public Boolean isRestrictedDelete()
```

Return Value
Type: Boolean

**isRestrictedPicklist()**
Returns `true` if the field is a restricted picklist, `false` otherwise.

**Signature**
```java
public Boolean isRestrictedPicklist()
```

Return Value
Type: Boolean

**isSearchPrefilterable()**
Returns `true` if a foreign key can be included in prefiltering when used in a SOSL WHERE clause, `false` otherwise.

**Signature**
```java
public Boolean isSearchPrefilterable()
```

Return Value
Type: Boolean

**Usage**
Prefiltering means to filter by a specific field value before executing the full search query. Prefiltering is supported only in WHERE clauses with the equals (==) operator.

**issSortable()**
Returns `true` if a query can sort on the field, `false` otherwise.
Signature
public Boolean isSortable()

Return Value
Type: Boolean

isUnique()
Returns true if the value for the field must be unique, false otherwise

Signature
public Boolean isUnique()

Return Value
Type: Boolean

isUpdateable()
Returns true if the field can be edited by the current user, or child records in a master-detail relationship field on a custom object can be reparented to different parent records; false otherwise.

⚠️ Important: Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Signature
public Boolean isUpdateable()

Return Value
Type: Boolean

isWriteRequiresMasterRead()
Returns true if writing to the detail object requires read sharing instead of read/write sharing of the parent.

Signature
public Boolean isWriteRequiresMasterRead()

Return Value
Type: Boolean

DescribelconResult Class
Contains icon metadata information for a tab.
Namespace

Schema

Usage

The `getIcons` method of the `Schema.DescribeTabResult` class returns a list of `Schema.DescribeIconResult` objects that describe colors used in a tab.

The methods in the `Schema.DescribeIconResult` class can be called using their property counterparts. For each method starting with `get`, you can omit the `get` prefix and the ending parentheses () to call the property counterpart. For example, `iconResultObj.url` is equivalent to `iconResultObj.getUrl()`.

Example

This sample shows how to get the icon information in the Sales app for the first tab's first icon.

```java
// Get tab set describes for each app
List<Schema.DescribeTabSetResult> tabSetDesc = Schema.describeTabs();

// Iterate through each tab set
for(Schema.DescribeTabSetResult tsr : tabSetDesc) {
    // Get tab info for the Sales app
    if (tsr.getLabel() == 'Sales') {
        // Get icon information for the first tab
        List<Schema.DescribeIconResult> iconDesc = tsr.getTabs()[0].getIcons();
        // Display the icon height and width of the first icon
        System.debug('Height: ' + iconDesc[0].getHeight());
        System.debug('Width: ' + iconDesc[0].getWidth());
    }
}
```

// Example debug statement output
// DEBUG|Height: 32
// DEBUG|Width: 32

DescribeIconResult Methods

The following are methods for `DescribeIconResult`. All are instance methods.

IN THIS SECTION:

- `getContentType()`
  Returns the tab icon’s content type, such as `image/png`.

- `getHeight()`
  Returns the tab icon’s height in pixels.

- `getTheme()`
  Returns the tab’s icon theme.

- `getUrl()`
  Returns the tab’s icon fully qualified URL.
getWidth()
Returns the tab’s icon width in pixels.

getContentType()
Returns the tab icon’s content type, such as image/png.

Signature
public String getContentType()

Return Value
Type: String

getHeight()
Returns the tab icon’s height in pixels.

Signature
public Integer getHeight()

Return Value
Type: Integer

Usage
Note: If the icon content type is SVG, the icon won’t have a size and its height is zero.

getTheme()
Returns the tab’s icon theme.

Signature
public String getTheme()

Return Value
Type: String
Possible theme values include theme3, theme4, and custom.
• theme3 is the Salesforce theme introduced during Spring ’10.
• theme4 is the Salesforce theme introduced in Winter ’14 for the mobile touchscreen version of Salesforce.
• custom is the theme name associated with a custom icon.
DescribeSObjectResult Class

Contains methods for describing SObjects. None of the methods take an argument.

Namespace

Schema

Usage

Note: If the icon content type is SVG, the icon won’t have a size and its width is zero.

DescribeSObjectResult Properties

The following are properties for DescribeSObjectResult.
**associateentitytype**

The type of associated object. For example, History or Share.

**Signature**

```java
public String associateentitytype {get; set;}
```

**Property Value**

Type: String

---

**associateparententity**

The parent object of an associated object.

**Signature**

```java
public String associateparententity {get; set;}
```

**Property Value**

Type: String

---

**DescribeSObjectResult Methods**

The following are methods for DescribeSObjectResult. All are instance methods.

**IN THIS SECTION:**

- **fields**
  - Follow fields with a field member variable name or with the `getMap` method.

- **fieldSets**
  - Follow fieldSets with a field set name or with the `getMap` method.

- **getAssociateEntityType()**
  - Returns additional metadata for an associated object of a specified parent but only if it’s a specific associated object type. Used in combination with the `getAssociateParentEntity()` method to get the parent object. For example, invoking the method on `AccountHistory` returns the parent object as `Account` and the type of associated object as `History`.

- **getAssociateParentEntity()**
  - Returns additional metadata for an associated object but only if it’s associated to a specific parent object. Used in combination with the `getAssociateEntityType()` method to get the type of associated object. For example, invoking the method on `AccountHistory` returns the parent object as `Account` and the type of associated object as `History`.

- **getChildRelationships()**
  - Returns a list of child relationships, which are the names of the sObjects that have a foreign key to the sObject being described.

- **getDefaultImplementation()**
  - Reserved for future use.

- **getHasSubtypes()**
  - Reserved for future use.
getImplementedBy()
Reserved for future use.

getImplementsInterfaces()
Reserved for future use.

getInterface()
Reserved for future use.

getKeyPrefix()
Returns the three-character prefix code for the object. Record IDs are prefixed with three-character codes that specify the type of the object (for example, accounts have a prefix of 001 and opportunities have a prefix of 006).

getLabel()
Returns the object's label, which may or may not match the object name.

getLabelPlural()
Returns the object's plural label, which may or may not match the object name.

getLocalName()
Returns the name of the object, similar to the getName method. However, if the object is part of the current namespace, the namespace portion of the name is omitted.

getName()
Returns the name of the object.

getRecordTypeInfos()
Returns a list of the record types supported by this object. The current user is not required to have access to a record type to see it in this list.

getRecordTypeInfosByDeveloperName()
Returns a map that matches developer names to their associated record type. The current user is not required to have access to a record type to see it in this map.

getRecordTypeInfosById()
Returns a map that matches record IDs to their associated record types. The current user is not required to have access to a record type to see it in this map.

getRecordTypeInfosByName()
Returns a map that matches record labels to their associated record type. The current user is not required to have access to a record type to see it in this map.

getSObjectDescribeOption()
Returns the effective describe option used by the system for the SObject.

getType()
Returns the Schema.SObjectType object for the sObject. You can use this to create a similar sObject.

isAccessible()
Returns true if the current user can see this object, false otherwise.

isCreateable()
Returns true if the object can be created by the current user, false otherwise.

isCustom()
Returns true if the object is a custom object, false if it is a standard object.
isCustomSetting()
Returns true if the object is a custom setting, false otherwise.

isDeletable()
Returns true if the object can be deleted by the current user, false otherwise.

isDeprecatedAndHidden()
Reserved for future use.

isFeedEnabled()
Returns true if Chatter feeds are enabled for the object, false otherwise. This method is only available for Apex classes and triggers saved using SalesforceAPI version 19.0 and later.

isMergeable()
Returns true if the object can be merged with other objects of its type by the current user, false otherwise. true is returned for leads, contacts, and accounts.

isMruEnabled()
Returns true if Most Recently Used (MRU) list functionality is enabled for the object, false otherwise.

isQueryable()
Returns true if the object can be queried by the current user, false otherwise.

isSearchable()
Returns true if the object can be searched by the current user, false otherwise.

isUndeletable()
Returns true if the object can be undeleted by the current user, false otherwise.

isUpdateable()
Returns true if the object can be updated by the current user, false otherwise.

fields
Follow fields with a field member variable name or with the getMap method.

Signature
public Schema.SObjectTypeFields fields()

Return Value
Type: The return value is a special data type. See the example to learn how to use fields.

Usage
Note: When you describe sObjects and their fields from within an Apex class, custom fields of new field types are returned regardless of the API version that the class is saved in. If a field type, such as the geolocation field type, is available only in a recent API version, components of a geolocation field are returned even if the class is saved in an earlier API version.

Example
To get a custom field name, specify the custom field name.

SEE ALSO:
   - Apex Developer Guide: Using Field Tokens
   - Apex Developer Guide: Describing sObjects Using Schema Method
   - Apex Developer Guide: Understanding Apex Describe Information

**fieldSets**

Follow `fieldSets` with a field set name or with the `getMap` method.

**Signature**

```java
public Schema.SObjectTypeFields fieldSets()
```

**Return Value**

Type: The return value is a special data type. See the example to learn how to use `fieldSets`.

**Example**

```java
Schema.DescribeSObjectResult d = Account.sObjectType.getDescribe();
Map<String, Schema.FieldSet> FsMap = d.fieldSets.getMap();
```

SEE ALSO:
   - Apex Developer Guide: Using Field Tokens
   - Apex Developer Guide: Describing sObjects Using Schema Method
   - Apex Developer Guide: Understanding Apex Describe Information

**getAssociateEntityType()**

Returns additional metadata for an associated object of a specified parent but only if it's a specific associated object type. Used in combination with the `getAssociateParentEntity()` method to get the parent object. For example, invoking the method on `AccountHistory` returns the parent object as `Account` and the type of associated object as `History`.

**Signature**

```java
public String associateentitytype {get; set;}
```

**Return Value**

Type: `String`

SEE ALSO:
   - DescribeSObjectResult Properties
getAssociateParentEntity()
Returns additional metadata for an associated object but only if it’s associated to a specific parent object. Used in combination with the getAssociateEntityType() method to get the type of associated object. For example, invoking the method on AccountHistory returns the parent object as Account and the type of associated object as History.

Signature
public String getAssociateParentEntity()

Return Value
Type: String

SEE ALSO:
DescribeSObjectResult Properties

getChildRelationships()
Returns a list of child relationships, which are the names of the sObjects that have a foreign key to the sObject being described.

Signature
public Schema.ChildRelationship getChildRelationships()

Return Value
Type: List<Schema.ChildRelationship>

Example
For example, the Account object includes Contacts and Opportunities as child relationships.

getDefaultImplementation()
Reserved for future use.

Signature
public String getDefaultImplementation()

Return Value
Type: String

getHasSubtypes()
Reserved for future use.
To check if Person Accounts are enabled for the current org, use this code snippet:
Schema.SObjectType.Account.fields.getMap().containsKey( 'isPersonAccount' );
Signature
public Boolean getHasSubtypes()

Return Value
Type: Boolean

getImplementedBy()
Reserved for future use.

Signature
public String getImplementedBy()

Return Value
Type: String

getImplementsInterfaces()
Reserved for future use.

Signature
public String getImplementsInterfaces()

Return Value
Type: String

getIsInterface()
Reserved for future use.

Signature
public Boolean getIsInterface()

Return Value
Type: Boolean

getKeyPrefix()
Returns the three-character prefix code for the object. Record IDs are prefixed with three-character codes that specify the type of the object (for example, accounts have a prefix of 001 and opportunities have a prefix of 006).
**Signature**

```
public String getKeyPrefix()
```

**Return Value**

Type: String

**Usage**

The DescribeSObjectResult object returns a value for objects that have a stable prefix. For object types that do not have a stable or predictable prefix, this field is blank. Client applications that rely on these codes can use this way of determining object type to ensure forward compatibility.

**getLabel()**

Returns the object's label, which may or may not match the object name.

**Signature**

```
public String getLabel()
```

**Return Value**

Type: String

**Usage**

The object's label might not always match the object name. For example, an organization in the medical industry might change the label for Account to Patient. This label is then used in the Salesforce user interface. See the Salesforce online help for more information.

**getLabelPlural()**

Returns the object's plural label, which may or may not match the object name.

**Signature**

```
public String getLabelPlural()
```

**Return Value**

Type: String

**Usage**

The object's plural label might not always match the object name. For example, an organization in the medical industry might change the plural label for Account to Patients. This label is then used in the Salesforce user interface. See the Salesforce online help for more information.
**getLocalName()**
Returns the name of the object, similar to the `getName` method. However, if the object is part of the current namespace, the namespace portion of the name is omitted.

**Signature**
```
public String getLocalName()
```

**Return Value**
Type: `String`

**getName()**
Returns the name of the object.

**Signature**
```
public String getName()
```

**Return Value**
Type: `String`

**getRecordTypeInfos()**
Returns a list of the record types supported by this object. The current user is not required to have access to a record type to see it in this list.

**Signature**
```
public List<Schema.RecordTypeInfo> getRecordTypeInfos()
```

**Return Value**
Type: `List<Schema.RecordTypeInfo>`

**getRecordTypeInfosByDeveloperName()**
Returns a map that matches developer names to their associated record type. The current user is not required to have access to a record type to see it in this map.

**Signature**
```
public Map<String, Schema.RecordTypeInfo> getRecordTypeInfosByDeveloperName()
```

**Return Value**
Type: `Map<String, Schema.RecordTypeInfo>`
**getRecordTypeInfosById()**
Returns a map that matches record IDs to their associated record types. The current user is not required to have access to a record type to see it in this map.

**Signature**

```java
public Schema.RecordTypeInfo getRecordTypeInfosById()
```

**Return Value**

Type: `Map<ID, Schema.RecordTypeInfo>`

**getRecordTypeInfosByName()**
Returns a map that matches record labels to their associated record type. The current user is not required to have access to a record type to see it in this map.

**Signature**

```java
public Schema.RecordTypeInfo getRecordTypeInfosByName()
```

**Return Value**

Type: `Map<String, Schema.RecordTypeInfo>`

**getSObjectDescribeOption()**
Returns the effective describe option used by the system for the SObject.

**Signature**

```java
public Schema.SObjectDescribeOptions getSObjectDescribeOption()
```

**Return Value**

Type: `Schema.SObjectDescribeOptions`

Valid values are:

- `SObjectDescribeOptions.FULL`: Indicates eager-load all elements of the describe, including child relationships, up-front at the time of method invocation.
- `SObjectDescribeOptions.DEFERRED`: Indicates lazy-load child relationships. This means that all child relationships will not be loaded at the time of first invocation of the method.

**getSObjectType()**
Returns the `Schema.SObjectType` object for the sObject. You can use this to create a similar sObject.

**Signature**

```java
public Schema.SObjectType getSObjectType()
```
Return Value
Type: Schema.SObjectType

**isAccessible()**
Returns `true` if the current user can see this object, `false` otherwise.

**Signature**
```java
public Boolean isAccessible()
```

**Return Value**
Type: Boolean

**Versioned Behavior Changes**
In API version 54.0 and later, for custom settings and custom metadata type objects,
DescribeSObjectResult.isAccessible() returns `false` if the user doesn't have permissions to access the queried objects. In API version 53.0 and earlier, the method returns `true` even if the user doesn't have the required permissions.

**isCreateable()**
Returns `true` if the object can be created by the current user, `false` otherwise.

**Signature**
```java
public Boolean isCreateable()
```

**Return Value**
Type: Boolean

**isCustom()**
Returns `true` if the object is a custom object, `false` if it is a standard object.

**Signature**
```java
public Boolean isCustom()
```

**Return Value**
Type: Boolean
Signature

public Boolean isCustomSetting()

Return Value

Type: Boolean

isDeletable()

Returns true if the object can be deleted by the current user, false otherwise.

Signature

public Boolean isDeletable()

Return Value

Type: Boolean

isDeprecatedAndHidden()

Reserved for future use.

Signature

public Boolean isDeprecatedAndHidden()

Return Value

Type: Boolean

isFeedEnabled()

Returns true if Chatter feeds are enabled for the object, false otherwise. This method is only available for Apex classes and triggers saved using SalesforceAPI version 19.0 and later.

Signature

public Boolean isFeedEnabled()

Return Value

Type: Boolean

isMergeable()

Returns true if the object can be merged with other objects of its type by the current user, false otherwise. true is returned for leads, contacts, and accounts.
Signature
public Boolean isMergeable()

Return Value
Type: Boolean

isMruEnabled()
Returns true if Most Recently Used (MRU) list functionality is enabled for the object, false otherwise.

Signature
public Boolean isMruEnabled()

Return Value
Type: Boolean

isQueryable()
Returns true if the object can be queried by the current user, false otherwise.

Signature
public Boolean isQueryable()

Return Value
Type: Boolean

isSearchable()
Returns true if the object can be searched by the current user, false otherwise.

Signature
public Boolean isSearchable()

Return Value
Type: Boolean

isUndeletable()
Returns true if the object can be undeleted by the current user, false otherwise.

Signature
public Boolean isUndeletable()
Return Value
Type: Boolean

isUpdateable()
Returns true if the object can be updated by the current user, false otherwise.

Signature
public Boolean isUpdateable()

Return Value
Type: Boolean

DescribeTabResult Class
Contains tab metadata information for a tab in a standard or custom app available in the Salesforce user interface.

Namespace
Schema

Usage
The getTabs method of the Schema.DescribeTabSetResult returns a list of Schema.DescribeTabResult objects that describe the tabs of one app.

The methods in the Schema.DescribeTabResult class can be called using their property counterparts. For each method starting with get, you can omit the get prefix and the ending parentheses () to call the property counterpart. For example, tabResultObj.label is equivalent to tabResultObj.getLabel(). Similarly, for each method starting with is, omit the is prefix and the ending parentheses (). For example, tabResultObj.isCustom is equivalent to tabResultObj.custom.

DescribeTabResult Methods
The following are methods for DescribeTabResult. All are instance methods.

IN THIS SECTION:
getColors()
Returns a list of color metadata information for all colors associated with this tab. Each color is associated with a theme and context.

getIconUrl()
Returns the URL for the main 32 x 32-pixel icon for a tab. This icon corresponds to the current theme (theme3) and appears next to the heading at the top of most pages.

getIcons()
Returns a list of icon metadata information for all icons associated with this tab. Each icon is associated with a theme and context.
getLabel()
Returns the display label of this tab.

getMiniIconUrl()
Returns the URL for the 16 x 16-pixel icon that represents a tab. This icon corresponds to the current theme (theme3) and appears in related lists and other locations.

getObjectName()
Returns the name of the sObject that is primarily displayed on this tab (for tabs that display a particular SOBject).

getUrl()
Returns a fully qualified URL for viewing this tab.

isCustom()
Returns true if this is a custom tab, or false if this is a standard tab.

gGetColors() (getColors)
Returns a list of color metadata information for all colors associated with this tab. Each color is associated with a theme and context.

Signature
public List<Schema.DescribeColorResult> getColors()

Return Value
Type: List<Schema.DescribeColorResult>

gGetIconUrl() (getIconUrl)
Returns the URL for the main 32 x 32-pixel icon for a tab. This icon corresponds to the current theme (theme3) and appears next to the heading at the top of most pages.

Signature
public String getIconUrl()

Return Value
Type: String

gGetIcons() (getIcons)
Returns a list of icon metadata information for all icons associated with this tab. Each icon is associated with a theme and context.

Signature
public List<Schema.DescribeIconResult> getIcons()

Return Value
Type: List<Schema.DescribeIconResult>
**getLabel()**

Returns the display label of this tab.

**Signature**

```java
public String getLabel()
```

**Return Value**

Type: `String`

**getMiniIconUrl()**

Returns the URL for the 16 x 16-pixel icon that represents a tab. This icon corresponds to the current theme (theme3) and appears in related lists and other locations.

**Signature**

```java
public String getMiniIconUrl()
```

**Return Value**

Type: `String`

**getSobjectName()**

Returns the name of the sObject that is primarily displayed on this tab (for tabs that display a particular SObject).

**Signature**

```java
public String getSobjectName()
```

**Return Value**

Type: `String`

**getUrl()**

Returns a fully qualified URL for viewing this tab.

**Signature**

```java
public String getUrl()
```

**Return Value**

Type: `String`
isCustom()
Returns true if this is a custom tab, or false if this is a standard tab.

Signature
public Boolean isCustom()

Return Value
Type: Boolean

DescribeTabSetResult Class
Contains metadata information about a Salesforce Classic standard or custom app available in the Salesforce user interface.

Namespace
Schema

Usage
The Schema.describeTabs method returns a list of Schema.DescribeTabSetResult objects that describe Salesforce Classic standard and custom apps.

The methods in the Schema.DescribeTabSetResult class can be called using their property counterparts. For each method starting with get, you can omit the get prefix and the ending parentheses () to call the property counterpart. For example, tabSetResultObj.label is equivalent to tabSetResultObj.getLabel(). Similarly, for each method starting with is, omit the is prefix and the ending parentheses (). For example, tabSetResultObj.isSelected is equivalent to tabSetResultObj.selected.

Example
This example shows how to call the Schema.describeTabs method to get describe information for all available Salesforce Classic apps. This example iterates through each describe result and gets more metadata information for the Sales app.

```java
// App we're interested to get more info about
String appName = 'Sales';

// Get tab set describes for each app
List<Schema.DescribeTabSetResult> tabSetDesc = Schema.describeTabs();

// Iterate through each tab set describe for each app and display the info
for (Schema.DescribeTabSetResult tsr : tabSetDesc) {
    // Get more information for the Sales app
    if (tsr.getLabel() == appName) {
        // Find out if the app is selected
        if (tsr.isSelected()) {
            System.debug('The ' + appName + ' app is selected.');
        }
        // Get the app's Logo URL and namespace
        String logo = tsr.getLogoUrl();
    }
}
```
System.debug('Logo URL: ' + logo);
String ns = tsr.getNamespace();
if (ns == '') {
    System.debug('The ' + appName + ' app has no namespace defined.');
} else {
    System.debug('Namespace: ' + ns);
}
// Get the number of tabs
System.debug('The ' + appName + ' app has ' + tsr.getTabs().size() + ' tabs.');

// Example debug statement output
// DEBUG|The Sales app is selected.
// DEBUG|Logo URL: https://MyDomainName.my.salesforce.com/img/seasonLogos/2014_winter_aloha.png
// DEBUG|The Sales app has no namespace defined.
// DEBUG|The Sales app has 14 tabs.

DescribeTabSetResult Methods

The following are methods for DescribeTabSetResult. All are instance methods.

IN THIS SECTION:

getDescription()
Returns the display description for the standard or custom app.

getLabel()
Returns the display label for the standard or custom app.

getLogoUrl()
Returns a fully qualified URL to the logo image associated with the standard or custom app.

getNamespace()
Returns the developer namespace prefix of a Salesforce AppExchange managed package.

getTabs()
Returns metadata information about the standard or custom app's displayed tabs.

isSelected()
Returns true if this standard or custom app is the user's currently selected app. Otherwise, returns false.

getDescription()
Returns the display description for the standard or custom app.

Signature

public String getDescription()
Return Value
Type: String

getLabel()
Returns the display label for the standard or custom app.

Signature
public String getLabel()

Return Value
Type: String

Usage
The display label changes when tabs are renamed in the Salesforce user interface. See the Salesforce online help for more information.

getLogoUrl()
Returns a fully qualified URL to the logo image associated with the standard or custom app.

Signature
public String getLogoUrl()

Return Value
Type: String

getAddress()
Returns the developer namespace prefix of a Salesforce AppExchange managed package.

Signature
public String getNamespace()

Return Value
Type: String

Usage
This namespace prefix corresponds to the namespace prefix of the Developer Edition organization that was enabled to allow publishing a managed package. This method applies to a custom app containing a set of tabs and installed as part of a managed package.
**getTabs()**
Returns metadata information about the standard or custom app's displayed tabs.

**Signature**
```
public List<Schema.DescribeTabResult> getTabs()
```

**Return Value**
Type: `List<Schema.DescribeTabResult>`

**isSelected()**
Returns `true` if this standard or custom app is the user's currently selected app. Otherwise, returns `false`.

**Signature**
```
public Boolean isSelected()
```

**Return Value**
Type: `Boolean`

### DisplayType Enum
A `Schema.DisplayType` enum value is returned by the field describe result's `getType` method.

**Namespace**
`Schema`

<table>
<thead>
<tr>
<th>Type Field Value</th>
<th>What the Field Object Contains</th>
</tr>
</thead>
<tbody>
<tr>
<td>address</td>
<td>Address values</td>
</tr>
<tr>
<td>anytype</td>
<td>Any value of the following types: <code>String</code>, <code>Picklist</code>, <code>Boolean</code>, <code>Integer</code>, <code>Double</code>, <code>Percent</code>, <code>ID</code>, <code>Date</code>, <code>DateTime</code>, <code>URL</code>, or <code>Email</code>.</td>
</tr>
<tr>
<td>base64</td>
<td>Base64-encoded arbitrary binary data (of type <code>base64Binary</code>)</td>
</tr>
<tr>
<td>Boolean</td>
<td>Boolean (<code>true</code> or <code>false</code>) values</td>
</tr>
<tr>
<td>Combobox</td>
<td>Comboboxes, which provide a set of enumerated values and allow the user to specify a value not in the list</td>
</tr>
<tr>
<td>Currency</td>
<td>Currency values</td>
</tr>
<tr>
<td>DataCategoryGroupReference</td>
<td>Reference to a data category group or a category unique name</td>
</tr>
<tr>
<td>Date</td>
<td>Date values</td>
</tr>
<tr>
<td>DateTime</td>
<td>DateTime values</td>
</tr>
<tr>
<td>Type Field Value</td>
<td>What the Field Object Contains</td>
</tr>
<tr>
<td>------------------</td>
<td>--------------------------------</td>
</tr>
<tr>
<td>Double</td>
<td>Double values</td>
</tr>
<tr>
<td>Email</td>
<td>Email addresses</td>
</tr>
<tr>
<td>EncryptedString</td>
<td>Encrypted string</td>
</tr>
<tr>
<td>ID</td>
<td>Primary key field for an object</td>
</tr>
<tr>
<td>Integer</td>
<td>Integer values</td>
</tr>
<tr>
<td>Location</td>
<td>Location values, including latitude and longitude.</td>
</tr>
<tr>
<td>Long</td>
<td>Long values</td>
</tr>
<tr>
<td>MultiPicklist</td>
<td>Multi-select picklists, which provide a set of enumerated values from which multiple values can be selected</td>
</tr>
<tr>
<td>Percent</td>
<td>Percent values</td>
</tr>
<tr>
<td>Phone</td>
<td>Phone numbers. Values can include alphabetic characters. Client applications are responsible for phone number formatting.</td>
</tr>
<tr>
<td>Picklist</td>
<td>Single-select picklists, which provide a set of enumerated values from which only one value can be selected</td>
</tr>
<tr>
<td>Reference</td>
<td>Cross-references to a different object, analogous to a foreign key field</td>
</tr>
<tr>
<td>String</td>
<td>String values</td>
</tr>
<tr>
<td>TextArea</td>
<td>String values that are displayed as multiline text fields</td>
</tr>
<tr>
<td>Time</td>
<td>Time values</td>
</tr>
<tr>
<td>URL</td>
<td>URL values that are displayed as hyperlinks</td>
</tr>
</tbody>
</table>

**Usage**

For more information, see Field Types in the Object Reference for Salesforce. For more information about the methods shared by all enums, see Enum Methods.

**FieldDescribeOptions Enum**

A Schema.FieldDescribeOptions enum value is a parameter in the SObjectType.getDescribe method.

**Usage**

For more information about the method using this enum, see getDescribe(options).

**Enum Values**

The following are the values of the Schema.FieldDescribeOptions enum.
<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEFAULT</td>
<td>Compute context-specific, describe field results.</td>
</tr>
<tr>
<td>FULL_DESCRIBE</td>
<td>Compute all aspects of describe field results.</td>
</tr>
</tbody>
</table>

### FieldSet Class

Contains methods for discovering and retrieving the details of field sets created on sObjects.

#### Namespace

*Schema*

#### Usage

Use the methods in the `Schema.FieldSet` class to discover the fields contained within a field set, and get details about the field set itself, such as the name, namespace, label, and so on. The following example shows how to get a collection of field set describe result objects for an sObject. The key of the returned Map is the field set name, and the value is the corresponding field set describe result.

```java
Map<String, Schema.FieldSet> FsMap =
    Schema.SObjectType.Account.fieldSets.getMap();
```

Field sets are also available from sObject describe results. The following lines of code are equivalent to the prior sample:

```java
Schema.DescribeSObjectResult d =
    Account.sObjectType.getDescribe();
Map<String, Schema.FieldSet> FsMap =
    d.fieldSets.getMap();
```

To work with an individual field set, you can access it via the map of field sets on an sObject or, when you know the name of the field set in advance, using an explicit reference to the field set. The following two lines of code retrieve the same field set:

```java
Schema.FieldSet fs1 = Schema.SObjectType.Account.fieldSets.getMap().get('field_set_name');
Schema.FieldSet fs2 = Schema.SObjectType.Account.fieldSets.field_set_name;
```

### Example: Displaying a Field Set on a Visualforce Page

This sample uses `Schema.FieldSet` and `Schema.FieldSetMember` methods to dynamically get all the fields in the Dimensions field set for the Merchandise custom object. The list of fields is then used to construct a SOQL query that ensures those fields are available for display. The Visualforce page uses the `MerchandiseDetails` class as its controller.

```java
public class MerchandiseDetails {
    public Merchandise__c merch { get; set; }

    public MerchandiseDetails() {
        this.merch = getMerchandise();
    }

    public List<Schema.FieldSetMember> getFields() {
```
private Merchandise__c getMerchandise() {
    String query = 'SELECT Id, Name FROM Merchandise__c LIMIT 1';
    return Database.query(query);
}

The Visualforce page using the above controller is simple:

One thing to note about the above markup is the expression used to determine if a field on the form should be indicated as being a required field. A field in a field set can be required by either the field set definition, or the field's own definition. The expression handles both cases.

FieldSet Methods

The following are methods for FieldSet. All are instance methods.

IN THIS SECTION:

    getDescription()
    Returns the field set's description.

    getFields()
    Returns a list of Schema.FieldSetMember objects for the fields making up the field set.

    getLabel()
    Returns the translation of the text label that is displayed next to the field in the Salesforce user interface.
**getSObjectType()**
Returns the `Schema.sObjectType` of the sObject containing the field set definition.

**getDescription()**
Returns the field set’s description.

**Signature**
```
public String getDescription()
```

**Return Value**
Type: `String`

**Usage**
Description is a required field for a field set, intended to describe the context and content of the field set. It’s often intended for administrators who might be configuring a field set defined in a managed package, rather than for end users.

**getFields()**
Returns a list of `Schema.FieldSetMember` objects for the fields making up the field set.

**Signature**
```
public List<FieldSetMember> getFields()
```

**Return Value**
Type: `List<Schema.FieldSetMember>`

**getLabel()**
Returns the translation of the text label that is displayed next to the field in the Salesforce user interface.

**Signature**
```
public String getLabel()
```

**Return Value**
Type: `String`
**getName()**

Returns the field set’s name.

**Signature**

```java
public String getName()
```

**Return Value**

Type: `String`

**getNamespace()**

Returns the field set’s namespace.

**Signature**

```java
public String getNamespace()
```

**Return Value**

Type: `String`

**Usage**

The returned namespace is an empty string if your organization hasn’t set a namespace, and the field set is defined in your organization. Otherwise, it’s the namespace of your organization, or the namespace of the managed package containing the field set.

**getSObjectType()**

Returns the `Schema.sObjectType` of the sObject containing the field set definition.

**Signature**

```java
public Schema.SObjectType getSObjectType()
```

**Return Value**

Type: `Schema.SObjectType`

---

**FieldSetMember Class**

Contains methods for accessing the metadata for field set member fields.

**Namespace**

`Schema`
Usage

Use the methods in the `Schema.FieldSetMember` class to get details about fields contained within a field set, such as the field label, type, a dynamic SOQL-ready field path, and so on. The following example shows how to get a collection of field set member describe result objects for a specific field set on an sObject:

```java
List<Schema.FieldSetMember> fields =
    Schema.SObjectType.Account.fieldSets.getMap().get('field_set_name').getFields();
```

If you know the name of the field set in advance, you can access its fields more directly using an explicit reference to the field set:

```java
List<Schema.FieldSetMember> fields =
    Schema.SObjectType.Account.fieldSets.field_set_name.getFields();
```

SEE ALSO:
- FieldSet Class

FieldSetMember Methods

The following are methods for `FieldSetMember`. All are instance methods.

IN THIS SECTION:
- `getDBRequired`
  Returns `true` if the field is required by the field's definition in its sObject, otherwise, `false`.
- `getFieldPath`
  Returns a field path string in a format ready to be used in a dynamic SOQL query.
- `getLabel`
  Returns the text label that's displayed next to the field in the Salesforce user interface.
- `getRequired`
  Returns `true` if the field is required by the field set, otherwise, `false`.
- `getType`
  Returns the field's Apex data type.
- `getSObjectField`
  Returns the token for this field.

`getDBRequired()`

Returns `true` if the field is required by the field's definition in its sObject, otherwise, `false`.

**Signature**

```java
public Boolean getDBRequired()
```

**Return Value**

Type: `Boolean`
getFieldPath()
Returns a field path string in a format ready to be used in a dynamic SOQL query.

Signature
public String getFieldPath()

Return Value
Type: String

Example
See Displaying a Field Set on a Visualforce Page for an example of how to use this method.

getLabel()
Returns the text label that’s displayed next to the field in the Salesforce user interface.

Signature
public String getLabel()

Return Value
Type: String

getRequired()
Returns true if the field is required by the field set, otherwise, false.

Signature
public Boolean getRequired()

Return Value
Type: Boolean

getType()
Returns the field’s Apex data type.

Signature
public Schema.DisplayType getType()

Return Value
Type: Schema.DisplayType
getSObjectField()  
Returns the token for this field.

Signature
public Schema.sObjectField getSObjectField()

Return Value
Type: Schema.SObjectField

PicklistEntry Class
Represents a picklist entry.

Namespace
Schema

Usage
Picklist fields contain a list of one or more items from which a user chooses a single item. They display as drop-down lists in the Salesforce user interface. One of the items can be configured as the default item.

A Schema.PicklistEntry object is returned from the field describe result using the getPicklistValues method. For example:

```java
Schema.DescribeFieldResult F = Account.Industry.getDescribe();
List<Schema.PicklistEntry> P = F.getPicklistValues();
```

PicklistEntry Methods
The following are methods for PicklistEntry. All are instance methods.

IN THIS SECTION:
- getLabel()  
  Returns the display name of this item in the picklist.
- getValue()  
  Returns the value of this item in the picklist.
- isActive()  
  Returns true if this item must be displayed in the drop-down list for the picklist field in the user interface, false otherwise.
- isDefaultValue()  
  Returns true if this item is the default value for the picklist, false otherwise. Only one item in a picklist can be designated as the default.
getLabel()
Returns the display name of this item in the picklist.

Signature
public String getLabel()

Return Value
Type: String

getValue()
Returns the value of this item in the picklist.

Signature
public String getValue()

Return Value
Type: String

isActive()
Returns true if this item must be displayed in the drop-down list for the picklist field in the user interface, false otherwise.

Signature
public Boolean isActive()

Return Value
Type: Boolean

isDefaultValue()
Returns true if this item is the default value for the picklist, false otherwise. Only one item in a picklist can be designated as the default.

Signature
public Boolean isDefaultValue()

Return Value
Type: Boolean
RecordTypeInfo Class

Contains methods for accessing record type information for an sObject with associated record types.

Namespace

Schema

Usage

A RecordTypeInfo object is returned from the sObject describe result using the `getRecordTypeInfos` method. For example:

```java
Schema.DescribeSObjectResult R = Account.SObjectType.getDescribe();
List<Schema.RecordTypeInfo> RT = R.getRecordTypeInfos();
```

In addition to the `getRecordTypeInfos` method, you can use the `getRecordTypeInfosById` and the `getRecordTypeInfosByName` methods. These methods return maps that associate RecordTypeInfo with record IDs and record labels, respectively.

Example

The following example assumes at least one record type has been created for the Account object:

```java
RecordType rt = [SELECT Id,Name FROM RecordType WHERE SobjectType='Account' LIMIT 1];
Map<Id,Schema.RecordTypeInfo> rtMapById = d.getRecordTypeInfosById();
Schema.RecordTypeInfo rtById = rtMapById.get(rt.id);
Map<String,Schema.RecordTypeInfo> rtMapByName = d.getRecordTypeInfosByName();
Schema.RecordTypeInfo rtByName = rtMapByName.get(rt.name);
System.assertEquals(rtById,rtByName);
```

RecordTypeInfo Methods

The following are methods for RecordTypeInfo. All are instance methods.

IN THIS SECTION:

- `getDeveloperName()`: Returns the developer name for this record type.
- `getName()`: Returns the UI label of this record type. The label can be translated into any language that Salesforce supports.
- `getRecordTypeId()`: Returns the ID of this record type.
- `isActive()`: Returns `true` if this record type is active, `false` otherwise.
- `isAvailable()`: Returns `true` if this record type is available to the current user, `false` otherwise. Use this method to display a list of available record types to the user when he or she is creating a new record.
isDefaultRecordTypeMapping()
Returns true if this is the default record type for the user, false otherwise.

isMaster()
Returns true if this is the master record type and false otherwise. The master record type is the default record type that’s used when a record has no custom record type associated with it.

getDeveloperName()
Returns the developer name for this record type.

Signature
public String getDeveloperName()

Return Value
Type: String

getName()
Returns the UI label of this record type. The label can be translated into any language that Salesforce supports.

Signature
public String getName()

Return Value
Type: String

getRecordTypeId()
Returns the ID of this record type.

Signature
public ID getRecordTypeId()

Return Value
Type: ID

isActive()
Returns true if this record type is active, false otherwise.

Signature
public Boolean isActive()
Return Value
Type: Boolean

isAvailable()
Returns true if this record type is available to the current user, false otherwise. Use this method to display a list of available record types to the user when he or she is creating a new record.

Signature
public Boolean isAvailable()

Return Value
Type: Boolean

isDefaultRecordTypeMapping()
Returns true if this is the default record type for the user, false otherwise.

Signature
public Boolean isDefaultRecordTypeMapping()

Return Value
Type: Boolean

isMaster()
Returns true if this is the master record type and false otherwise. The master record type is the default record type that’s used when a record has no custom record type associated with it.

Signature
public Boolean isMaster()

Return Value
Type: Boolean

SOAPType Enum
A Schema.SOAPType enum value is returned by the field describe result getSoapType method.

Namespace
Schema
### Type Field Value

<table>
<thead>
<tr>
<th>Type Field Value</th>
<th>What the Field Object Contains</th>
</tr>
</thead>
<tbody>
<tr>
<td>anytype</td>
<td>Any value of the following types: <em>String</em>, <em>Boolean</em>, <em>Integer</em>, <em>Double</em>, <em>ID</em>, <em>Date</em> or <em>DateTime</em>.</td>
</tr>
<tr>
<td>base64binary</td>
<td>Base64-encoded arbitrary binary data (of type base64Binary)</td>
</tr>
<tr>
<td>Boolean</td>
<td>Boolean (<em>true</em> or <em>false</em>) values</td>
</tr>
<tr>
<td>Date</td>
<td>Date values</td>
</tr>
<tr>
<td>DateTime</td>
<td>DateTime values</td>
</tr>
<tr>
<td>Double</td>
<td>Double values</td>
</tr>
<tr>
<td>ID</td>
<td>Primary key field for an object</td>
</tr>
<tr>
<td>Integer</td>
<td>Integer values</td>
</tr>
<tr>
<td>String</td>
<td>String values</td>
</tr>
<tr>
<td>Time</td>
<td>Time values</td>
</tr>
</tbody>
</table>

### Usage

For more information, see [SOAPTypes](https://documentation.salesforce.com/soapapi/soapapi.htm#SOAPTypes) in the *SOAP API Developer Guide*. For more information about the methods shared by all enums, see [Enum Methods](https://developer.salesforce.com/docs/atlas.en-us.ipick.doapick.meta/ipick.doapick.htm).

### SObjectDescribeOptions Enum

A `Schema.SObjectDescribeOptions` enum value is a parameter in the `SObjectType.getDescribe()` method.

#### Usage

For more information about the method using this enum, see `getDescribe(options)`.

#### Enum Values

The following are the values of the `Schema.SObjectDescribeOptions` enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEFAULT</td>
<td>Either eager-load or lazy-load depending on the API version.</td>
</tr>
<tr>
<td>DEFERRED</td>
<td>Lazy-load child relationships; do not load all child relationships at the time of first invocation of the method.</td>
</tr>
<tr>
<td>FULL</td>
<td>Eager-load all elements of the describe, including child relationships, up-front at the time of method invocation.</td>
</tr>
</tbody>
</table>

See `getDescribe(options)`. 
SOBJECTFIELD Class

A Schema.sObjectField object is returned from the field describe result using the getController and getSObjectField methods.

Namespace

Schema

Example

```java
Schema.DescribeFieldResult F = Account.Industry.getDescribe();
Schema.sObjectField T = F.getSObjectField();
```

sObjectField Methods

The following are instance methods for sObjectField.

IN THIS SECTION:

- `getDescribe()`
  - Returns the describe field result for this field.
- `getDescribe(options)`
  - Returns the describe field result for this field. This method also provides an option to get all the describe field results for an object.

`getDescribe()`

Returns the describe field result for this field.

**Signature**

```java
public Schema.DescribeFieldResult getDescribe()
```

**Return Value**

Type: Schema.DescribeFieldResult

`getDescribe(options)`

Returns the describe field result for this field. This method also provides an option to get all the describe field results for an object.

**Signature**

```java
public Schema.DescribeFieldResult getDescribe(Object options)
```

**Parameters**

- `options`  
  Type: Object
Use this parameter to pass FieldDescribeOptions.FULL_DESCRIBE when a subset of system objects could have different results for picklist values based on the context they're invoked in. This parameter computes all aspects of describe field results.

For example, AICConversationContext.PersonType field is a picklist that contains a list of accessible object types.

**Return Value**

Type: Schema.DescribeFieldResult

### SObjectType Class

A Schema.sObjectType object is returned from the field describe result using the getReferenceTo method, or from the sObject describe result using the getSObjectType method.

### Namespace

Schema

### Usage

```java
Schema.DescribeFieldResult F = Account.Industry.getDescribe();
List<Schema.sObjectType> P = F.getReferenceTo();
```

### SObjectType Methods

The following are methods for SObjectType. All are instance methods.

**IN THIS SECTION:**

- `getDescribe()`
  
  Returns the describe sObject result for this field.

- `getDescribe(options)`
  
  Returns the describe sObject result for this field; the parameter value determines whether all child relationships are loaded up-front, or not.

- `newSObject()`
  
  Constructs a new sObject of this type.

- `newSObject(id)`
  
  Constructs a new sObject of this type, with the specified ID.

- `newSObject(recordTypeId, loadDefaults)`
  
  Constructs a new sObject of this type, and optionally, of the specified record type ID and with default custom field values.

**getDescribe()**

Returns the describe sObject result for this field.
Signature

```java
public Schema.DescribeSObjectResult getDescribe()
```

Return Value
Type: `Schema.DescribeSObjectResult`

**getDescribe(options)**
Returns the describe sObject result for this field; the parameter value determines whether all child relationships are loaded up-front, or not.

Signature

```java
public Schema.DescribeSObjectResult getDescribe(Object options)
```

Parameters

- `options`
  Type: `Object`
  The parameter values determine how the elements of the describe operation are loaded.
  - Use `SObjectDescribeOptions.FULL` to eager-load all elements of the describe, including child relationships, up-front at the time of method invocation. This describe guarantees fully coherent results, even if the describe object is passed to another namespace, API version, or other Apex context that may have different results when generating describe attributes.
  - Use `SObjectDescribeOptions.DEFERRED` to enable lazy initialization of describe attributes on first use. This means that all child relationships will not be loaded at the time of first invocation of the method.
  - Use `SObjectDescribeOptions.DEFAULT` to default to either eager-load or lazy-load depending on the API version.

The type of describe operation, as determined by the parameter value is depicted in this table.

<table>
<thead>
<tr>
<th>Parameter Value</th>
<th>API Version 43.0 and Earlier</th>
<th>API Version 44.0 and Later</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full</td>
<td>Eager</td>
<td>Eager</td>
</tr>
<tr>
<td>Deferred</td>
<td>Lazy</td>
<td>Lazy</td>
</tr>
<tr>
<td>Default</td>
<td>Lazy</td>
<td>Lazy</td>
</tr>
</tbody>
</table>

Return Value
Type: `Schema.DescribeSObjectResult`

**newSObject()**
Constructs a new sObject of this type.

Signature

```java
public sObject newSObject()
```
Return Value
Type: sObject

Example
For an example, see Dynamic DML.

newSObject(id)
Constructs a new sObject of this type, with the specified ID.

Signature
public sObject newSObject(ID id)

Parameters
id
Type: ID

Return Value
Type: sObject

Usage
For the argument, pass the ID of an existing record in the database.
After you create a new sObject, the sObject returned has all fields set to null. You can set any updateable field to desired values and then update the record in the database. Only the fields you set new values for are updated and all other fields which are not system fields are preserved.

newSObject(recordTypeId, loadDefaults)
Constructs a new sObject of this type, and optionally, of the specified record type ID and with default custom field values.

Important: Where possible, we changed noninclusive terms to align with our company value of Equality. We maintained certain terms to avoid any effect on customer implementations.

Signature
public sObject newSObject(ID recordTypeId, Boolean loadDefaults)

Parameters
recordTypeId
Type: ID
Specifies the record type ID of the sObject to create. If no record type exists for this sObject, use null. If the sObject has record types and you specify null, the default record type is used.
loadDefaults
Type: Boolean
Specifies whether to populate custom fields with their predefined default values (true) or not (false).

Return Value
Type: sObject

Usage
• For required fields that have no default values, make sure to provide a value before inserting the new sObject. Otherwise, the insertion results in an error. An example is the Account Name field or a master-detail relationship field.
• Since picklists and multi-select picklists can have default values specified per record type, this method populates the default value corresponding to the record type specified.
• If fields have no predefined default values and the loadDefaults argument is true, this method creates the sObject with field values of null.
• If the loadDefaults argument is false, this method creates the sObject with field values of null.
• This method populates read-only custom fields of the new sObject with default values. You can then insert the new sObject with the read-only fields, even though these fields cannot be edited after they’re inserted.
• If a custom field is marked as unique and also provides a default value, inserting more than one new sObject will cause a run-time exception because of duplicate field values.

To learn more about default field values, see “Default Field Values” in the Salesforce online help.

Example: Creating New sObject with Default Values
This sample creates an account with any default values populated for its custom fields, if any, using the newSObject method. It also creates a second account for a specific record type. For both accounts, the sample sets the Name field, which is a required field that doesn’t have a default value, before inserting the new accounts.

```apex
// Create an account with predefined default values
Account acct = (Account)Account.sObjectType.newSObject(null, true);
// Provide a value for Name
acct.Name = 'Acme';
// Insert new account
insert acct;

// This is for record type RT1 of Account
ID rtId = [SELECT Id FROM RecordType WHERE sObjectType='Account' AND Name='RT1'].Id;
Account acct2 = (Account)Account.sObjectType.newSObject(rtId, true);
// Provide a value for Name
acct2.Name = 'Acme2';
// Insert new account
insert acct2;
```

Search Namespace

The Search namespace provides classes for getting search results and suggestion results.
The following are the classes in the Search namespace.
IN THIS SECTION:

KnowledgeSuggestionFilter Class
Filter settings that narrow the results from a call to System.Search.suggest(searchQuery, sObjectType, options) when the SOSL search query contains a KnowledgeArticleVersion object.

QuestionSuggestionFilter Class
The Search.QuestionSuggestionFilter class filters results from a call to System.Search.suggest(searchQuery, sObjectType, options) when the SOSL searchQuery contains a FeedItem object.

SearchResult Class
A wrapper object that contains an sObject and search metadata.

SearchResults Class
Wraps the results returned by the Search.find(String) method.

SuggestionOption Class
Options that narrow record and article suggestion results returned from a call to System.Search.suggest(String, String, Search.SuggestionOption).

SuggestionResult Class
A wrapper object that contains an sObject.

SuggestionResults Class
Wraps the results returned by the Search.suggest(String, String, Search.SuggestionOption) method.

SEE ALSO:

find(searchQuery)
suggest(searchQuery, sObjectType, suggestions)

KnowledgeSuggestionFilter Class
Filter settings that narrow the results from a call to System.Search.suggest(searchQuery, sObjectType, options) when the SOSL search query contains a KnowledgeArticleVersion object.

Namespace
Search

KnowledgeSuggestionFilter Methods
The following are methods for KnowledgeSuggestionFilter.

IN THIS SECTION:

addArticleType(articleType)
Adds a filter that narrows suggestion results to display the specified article type. This filter is optional.

addDataCategory(dataCategoryGroupName, dataCategoryName)
Adds a filter that narrows suggestion results to display articles in the specified data category. This filter is optional.
addTopic(topic)
Specifies the article topic to return. This filter is optional.

setChannel(channelName)
Sets a channel to narrow the suggestion results to articles in the specified channel. This filter is optional.

setDataCategories(dataCategoryFilters)
Adds filters that narrow suggestion results to display articles in the specified data categories. Use this method to set multiple data category group and name pairs in one call. This filter is optional.

setLanguage(localeCode)
Sets a language to narrow the suggestion results to display articles in that language. This filter value is required in calls to System.Search.suggest(String, String, Search.SuggestionOption).

setPublishStatus(publishStatus)
Sets a publish status to narrow the suggestion results to display articles with that status. This filter value is required in calls to System.Search.suggest(String, String, Search.SuggestionOption).

setValidationStatus(validationStatus)
Sets a validation status to narrow the suggestion results to display articles with that status. This filter is optional.

addArticleType(articleType)
Adds a filter that narrows suggestion results to display the specified article type. This filter is optional.

**Signature**

public void addArticleType(String articleType)

**Parameters**

*articleType*
  
<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>A three-character ID prefix indicating the desired article type.</td>
</tr>
</tbody>
</table>

**Return Value**

Type: void

**Usage**

To add more than 1 article type, call the method multiple times.

addDataCategory(dataCategoryGroupName, dataCategoryName)
Adds a filter that narrows suggestion results to display articles in the specified data category. This filter is optional.

**Signature**

public void addDataCategory(String dataCategoryGroupName, String dataCategoryName)
Parameters

dataCategoryGroupName
  Type: String
  The name of the data category group

dataCategoryName
  Type: String
  The name of the data category.

Return Value
Type: void

Usage
To set multiple data categories, call the method multiple times. The name of the data category group and name of the data category for desired articles, expressed as a mapping, for example,
Search.KnowledgeSuggestionFilter.addDataCategory('Regions', 'Asia').

addTopic(topic)
Specifies the article topic to return. This filter is optional.

Signature
public void addTopic(String topic)

Parameters
addTopic
  Type: String
  The name of the article topic.

Return Value
Type: void

Usage
To add more than 1 article topic, call the method multiple times.

setChannel(channelName)
Sets a channel to narrow the suggestion results to articles in the specified channel. This filter is optional.

Signature
public void setChannel(String channelName)
Parameters

channelName
  Type: String
  The name of a channel. Valid values are:
  • AllChannels—Visible in all channels the user has access to
  • App—Visible in the internal Salesforce Knowledge application
  • Pkb—Visible in the public knowledge base
  • Csp—Visible in the Customer Portal
  • Prm—Visible in the Partner Portal
  If channel isn’t specified, the default value is determined by the type of user.
  • Pkb for a guest user
  • Csp for a Customer Portal user
  • Prm for a Partner Portal user
  • App for any other type of user
  If channel is specified, the specified value may not be the actual value requested, because of certain requirements.
  • For guest, Customer Portal, and Partner Portal users, the specified value must match the default value for each user type. If the values don’t match or AllChannels is specified, then App replaces the specified value.
  • For all users other than guest, Customer Portal, and Partner Portal users:
    – If Pkb, Csp, Prm, or App are specified, then the specified value is used.
    – If AllChannels is specified, then App replaces the specified value.

Return Value

Type: void

setDataCategories(dataCategoryFilters)

Adds filters that narrow suggestion results to display articles in the specified data categories. Use this method to set multiple data category group and name pairs in one call. This filter is optional.

Signature

public void setDataCategories(Map dataCategoryFilters)

Parameters

dataCategoryFilters
  Type: Map
  A map of data category group and data category name pairs.

Return Value

Type: void
**setLanguage(localeCode)**

Sets a language to narrow the suggestion results to display articles in that language. This filter value is required in calls to `System.Search.suggest(String, String, Search.SuggestionOption)`.

**Signature**

```java
public void setLanguage(String localeCode)
```

**Parameters**

- `localeCode`
  - Type: `String`
  - A locale code. For example, 'en_US' (English–United States), or 'es' (Spanish).

**Return Value**

Type: void

**SEE ALSO:**

- Supported Locales

**setPublishStatus(publishStatus)**

Sets a publish status to narrow the suggestion results to display articles with that status. This filter value is required in calls to `System.Search.suggest(String, String, Search.SuggestionOption)`.

**Signature**

```java
public void setPublishStatus(String publishStatus)
```

**Parameters**

- `publishStatus`
  - Type: `String`
  - A publish status. Valid values are:
    - Draft—Articles aren’t published in Salesforce Knowledge.
    - Online—Articles are published in Salesforce Knowledge.
    - Archived—Articles aren’t published and are available in Archived Articles view.

**setValidationStatus(validationStatus)**

Sets a validation status to narrow the suggestion results to display articles with that status. This filter is optional.

**Signature**

```java
public void setValidationStatus(String validationStatus)
```
Parameters

validationStatus
Type: String
An article validation status. These values are available in the ValidationStatus field on the KnowledgeArticleVersion object.

Return Value
Type: void

QuestionSuggestionFilter Class
The Search.QuestionSuggestionFilter class filters results from a call to System.Search.suggest(searchQuery, sObjectType, options) when the SOSL searchQuery contains a FeedItem object.

Namespace
Search

IN THIS SECTION:
QuestionSuggestionFilter Methods

QuestionSuggestionFilter Methods
The following are methods for QuestionSuggestionFilter.

IN THIS SECTION:
addGroupId(groupId)
Adds a filter to display questions associated with the single specified group whose ID is passed in as an argument. This filter is optional.
addNetworkId(networkId)
Adds a filter to display questions associated with the single specified network whose ID is passed in as an argument. This filter is optional.
addUserId(userId)
Adds a filter to display questions belonging to the single specified user whose ID is passed in as an argument. This filter is optional.
setGroupIds(groupIds)
Sets a new list of groups to replace the current list of groups where the group IDs are passed in as an argument. This filter is optional.
setNetworkIds(networkIds)
Sets a new list of networks to replace the current list of networks where the network IDs are passed in as an argument. This filter is optional.
setTopicId(topicId)
Sets a filter to display questions associated with the single specified topic whose ID is passed in as an argument. This filter is optional.
setUserIds(userIds)
Sets a new list of users to replace the current list of users where the users IDs are passed in as an argument. This filter is optional.
addGroupId(groupId)

Adds a filter to display questions associated with the single specified group whose ID is passed in as an argument. This filter is optional.

Signature

public void addGroupId(String groupId)

Parameters

groupId

Type: String

The ID for a group.

Return Value

Type: void

Usage

To add more than one group, call the method multiple times.

addNetworkId(networkId)

Adds a filter to display questions associated with the single specified network whose ID is passed in as an argument. This filter is optional.

Signature

public void addNetworkId(String networkId)

Parameters

networkId

Type: String

The ID of the Experience Cloud site about which you’re retrieving this information.

Return Value

Type: void

Usage

To add more than one network, call the method multiple times.

addUserId(userId)

Adds a filter to display questions belonging to the single specified user whose ID is passed in as an argument. This filter is optional.

Signature

public void addUserId(String userId)
Parameters

`userId`
  Type: `String`
  The ID for the user.

Return Value

Type: `void`

Usage

To add more than one user, call the method multiple times.

`setGroupIds(groupIds)`

Sets a new list of groups to replace the current list of groups where the group IDs are passed in as an argument. This filter is optional.

Signature

`public void setGroupIds(List<String> groupIds)`

Parameters

`groupIds`
  Type: `List<String>`
  A list of group IDs.

Return Value

Type: `void`

`setNetworkIds(networkIds)`

Sets a new list of networks to replace the current list of networks where the network IDs are passed in as an argument. This filter is optional.

Signature

`public void setNetworkIds(List<String> networkIds)`

Parameters

`networkIds`
  Type: `List<String>`
  A list of network IDs.

Return Value

Type: `void`
**setTopicId(topicId)**
Sets a filter to display questions associated with the single specified topic whose ID is passed in as an argument. This filter is optional.

**Signature**
```
public void setTopicId(String topicId)
```

**Parameters**
- **topicId**
  Type: `String`  
The ID for a topic.

**Return Value**
Type: `void`

**setUserIds(userIds)**
Sets a new list of users to replace the current list of users where the users IDs are passed in as an argument. This filter is optional.

**Signature**
```
public void setUserIds(List<String> userIds)
```

**Parameters**
- **userIds**
  Type: `List<String>`  
  A list of user IDs.

**Return Value**
Type: `void`

**SearchResult Class**
A wrapper object that contains an sObject and search metadata.

**Namespace**
`Search`

**SearchResult Methods**
The following are methods for `SearchResult`. 
IN THIS SECTION:

- **getSObject()**
  Returns an sObject from a SearchResult object.

- **getSnippet(fieldName)**
  Returns a snippet from a Case, Feed, or Knowledge Article SearchResult object based on the specified field name.

- **getSnippet()**
  Returns a snippet from a SearchResult object based on the default field.

### getSObject()

Returns an sObject from a SearchResult object.

**Signature**

```java
public SObject getSObject()
```

**Return Value**

Type: SObject

**SEE ALSO:**

- **find(searchQuery)**
  Apex Developer Guide: Dynamic SOSL

### getSnippet(fieldName)

Returns a snippet from a Case, Feed, or Knowledge Article SearchResult object based on the specified field name.

**Signature**

```java
public String getSnippet(String fieldName)
```

**Parameters**

- **fieldName**
  Type: String

  The field name to use for creating the snippet.

  Valid values: Case.Casenumber, FeedPost.Title, KnowledgeArticleVersion.Title

**Return Value**

Type: String

**SEE ALSO:**

- **find(searchQuery)**
  Apex Developer Guide: Dynamic SOSL
getSnippet()  
Returns a snippet from a SearchResult object based on the default field.

Signature

public String getSnippet()

Return Value

Type: String

SEE ALSO:

- find(searchQuery)
- Apex Developer Guide: Dynamic SOSL

SearchResults Class

Wraps the results returned by the Search.find(String) method.

Namespace

Search

SearchResults Methods

The following are methods for SearchResults.

IN THIS SECTION:

- get(sObjectType)
  - Returns a list of Search.SearchResult objects that contain an sObject of the specified type.

get(sObjectType)

Returns a list of Search.SearchResult objects that contain an sObject of the specified type.

Signature

public List<Search.SearchResult> get(String sObjectType)

Parameters

sObjectType
  - Type: String

  The name of an sObject in the dynamic SOSL query passed to the Search.find(String) method.
Return Value
Type: List<Search.SearchResult>

Usage
SOSL queries passed to the Search.find(String) method can return results for multiple objects. For example, the query
Search.find('FIND map IN ALL FIELDS RETURNING Account, Contact, Opportunity') includes results for 3 objects. You can call get(string) to retrieve search results for 1 object at a time. For example, to get results for the Account object, call Search.SearchResults.get('Account').

SEE ALSO:
- find(searchQuery)
- SearchResult Methods
- Apex Developer Guide: Dynamic SOSL

SuggestionOption Class
Options that narrow record and article suggestion results returned from a call to System.Search.suggest(String, String, Search.SuggestionOption).

Namespace
Search

SuggestionOption Methods
The following are methods for SuggestionOption.

IN THIS SECTION:
- setFilter(knowledgeSuggestionFilter)
  Set filters that narrow Salesforce Knowledge article results in a call to System.Search.suggest(String, String, Search.SuggestionOption).
- setLimit(limit)
  The maximum number of record or article suggestions to retrieve.

setFilter (knowledgeSuggestionFilter)
Set filters that narrow Salesforce Knowledge article results in a call to System.Search.suggest(String, String, Search.SuggestionOption).

Signature
public void setFilter(Search.KnowledgeSuggestionFilter knowledgeSuggestionFilter)
Parameters

knowledgeSuggestionFilter
Type: KnowledgeSuggestionFilter
An object containing filters that narrow the search results.

Return Value
Type: void

Usage

Search.KnowledgeSuggestionFilter filters = new Search.KnowledgeSuggestionFilter();
filters.setLanguage('en_US');
filters.setPublishStatus('Online');
filters.setChannel('app');

Search.SuggestionOption options = new Search.SuggestionOption();
options.setFilter(filters);

Search.SuggestionResults suggestionResults = Search.suggest('all', 'KnowledgeArticleVersion',
options);

for (Search.SuggestionResult searchResult : suggestionResults.getSuggestionResults()) {
    KnowledgeArticleVersion article = (KnowledgeArticleVersion) searchResult.getSObject();
    System.debug(article.title);
}

setLimit(limit)
The maximum number of record or article suggestions to retrieve.

Signature

public void setLimit(Integer limit)

Parameters

limit
Type: Integer
The maximum number of record or article suggestions to retrieve.

Return Value
Type: void

Usage

By default, the System.Search.suggest(String, String, Search.SuggestionOption) method returns the
5 most relevant results. However, if your query is broad, it could match more than 5 results. If
Search.SuggestionResults.hasMoreResults() returns true, there are more than 5 results. To retrieve them, call setLimit(Integer) to increase the number of suggestions results.

```java
Search.SuggestionOption option = new Search.SuggestionOption();
option.setLimit(10);
Search.suggest('my query', 'mySObjectType', option);
```

**SuggestionResult Class**

A wrapper object that contains an sObject.

**Namespace**

Search

**SuggestionResult Methods**

The following are methods for SuggestionResult.

IN THIS SECTION:

getSObject()

Returns the sObject from a SuggestionResult object.

**getSObject()**

Returns the sObject from a SuggestionResult object.

**Signature**

```java
public SObject getSObject()
```

**Return Value**

Type: SObject

**SuggestionResults Class**

Wraps the results returned by the Search.suggest(String, String, Search.SuggestionOption) method.

**Namespace**

Search

**SuggestionResults Methods**

The following are methods for SuggestionResults.
getSuggestionResults()
Returns a list of SuggestionResult objects from the response to a call to Search.suggest(String, String, Search.SuggestionOption).

hasMoreResults()
Indicates whether a call to System.Search.suggest(String, String, Search.SuggestionOption) has more results available than were returned.

g getSuggestionResults()
Returns a list of SuggestionResult objects from the response to a call to Search.suggest(String, String, Search.SuggestionOption).

Signature
public List<Search.SuggestionResult> getSuggestionResults()

Return Value
Type: List<SuggestionResult>

hasMoreResults()
Indicates whether a call to System.Search.suggest(String, String, Search.SuggestionOption) has more results available than were returned.

Signature
public Boolean hasMoreResults()

Return Value
Type: Boolean

Usage
If a limit isn’t specified, 5 records are returned in calls to System.Search.suggest(String, String, Search.SuggestionOption). If there are more suggested records than the limit specified, a call to hasMoreResults() returns true.

Sfc Namespace

The Sfc namespace contains classes used in Salesforce Files.
The following are the classes in the Sfc namespace.
ContentDownloadContext Enum

This enum specifies the download context.

Usage

If the operationContext is CONTENT, CHATTER, DELIVERY, S1, or MOBILE, it can be used in a shepherd servlet as a query parameter. It's possible for a user to change the query parameters. If a user enters a value other than CONTENT, CHATTER, DELIVERY, S1, or MOBILE, the value is treated as the default value CONTENT.

Users can't set query parameters to REST_API, SOQL, or RETRIEVE, so these values can be assumed to be accurate.

Enum Values

The Sfc.ContentDownloadContext enum value identifies the content download context. The enum value is provided as a query parameter in the file download servlet. The following are the values of the Sfc.ContentDownloadContext enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHATTER</td>
<td>Download from Chatter.</td>
</tr>
<tr>
<td>CONTENT</td>
<td>Default value. Downloads from the Salesforce CRM Content product.</td>
</tr>
<tr>
<td>DELIVERY</td>
<td>Download of a content delivery.</td>
</tr>
<tr>
<td>REST_API</td>
<td>Download from the Connect API(/connect/files/${fileId}/content endpoint). Used in both Android and iOS apps.</td>
</tr>
<tr>
<td>RETRIEVE</td>
<td>Retrieve VersionData from SObject API.</td>
</tr>
<tr>
<td>S1</td>
<td>Download from Lightning Experience.</td>
</tr>
<tr>
<td>SOQL</td>
<td>Select VersionData from SOQL.</td>
</tr>
</tbody>
</table>

ContentDownloadHandler Class

Use ContentDownloadHandler to define a custom download handler that controls how content is downloaded.

Namespace

Sfc on page 2785
ContentDownloadHandler Properties

The following are properties for ContentDownloadHandler.

**downloadErrorMessage**
A customized error message explaining why the download isn’t allowed.

**isDownloadAllowed**
Indicates whether or not download is allowed.

**redirectUrl**
The URL the user should be redirected to, for applying Information Rights Management (IRM) control, virus scanning, or other behavior.

**downloadErrorMessage**
A customized error message explaining why the download isn’t allowed.

**Signature**
```java
public String downloadErrorMessage {get; set;}
```

**Property Value**
- **Type:** String
- This message is used if a redirectUrl is not provided. If the download is not allowed, Salesforce will throw a ContentCustomizedDownloadException exception that contains the downloadErrorMessage.

**isDownloadAllowed**
Indicates whether or not download is allowed.

**Signature**
```java
public Boolean isDownloadAllowed {get; set;}
```

**Property Value**
- **Type:** Boolean

**redirectUrl**
The URL the user should be redirected to, for applying Information Rights Management (IRM) control, virus scanning, or other behavior.
Signature

public String redirectUrl {get; set;}

Property Value

Type: String

The URL must be a valid relative URL. For example, the redirect can be a custom Visualforce page such as "/apex/IRMControl". URLs with no path, such as "www.domain.com", will result in an InvalidParameterValueException.

ContentDownloadHandlerFactory Interface

Use this interface to provide a class factory that Salesforce can call to create instances of your custom ContentDownloadHandler.

Namespace

Sfc on page 2785

Usage

ContentDownloadHandler getContentDownloadHandler(List<ID> ids, ContentDownloadContext context);

ContentDownloadHandlerFactory Methods

The following are methods for ContentDownloadHandlerFactory.

getContentDownloadHandler(var1, var2)

Returns a ContentDownloadHandler for a given list of content IDs and a download context.

getContentDownloadHandler(var1, var2)

Returns a ContentDownloadHandler for a given list of content IDs and a download context.

Signature

public Sfc.ContentDownloadHandler getContentDownloadHandler(List<Id> var1, Sfc.ContentDownloadContext var2)

Parameters

var1

Type: List<Id>
ContentDownloadHandlerFactory Example Implementation

This example creates a class that implements the `Sfc.ContentDownloadHandlerFactory` interface and returns a download handler that blocks downloading content to mobile devices.

```java
// Allow customization of the content download experience
public class ContentDownloadHandlerFactoryImpl implements Sfc.ContentDownloadHandlerFactory {

    public Sfc.ContentDownloadHandler getContentDownloadHandler(List<ID> ids, Sfc.ContentDownloadContext context) {
        Sfc.ContentDownloadHandler contentDownloadHandler = new Sfc.ContentDownloadHandler();

        if (context == Sfc.ContentDownloadContext.MOBILE) {
            contentDownloadHandler.isDownloadAllowed = false;
            contentDownloadHandler.downloadErrorMessage = 'Downloading a file from a mobile device is not allowed.';
            return contentDownloadHandler;
        }

        contentDownloadHandler.isDownloadAllowed = true;
        return contentDownloadHandler;
    }
}
```

### Sfdc_Checkout Namespace

The `Sfdc_Checkout` namespace provides an interface and classes for B2B Commerce apps in Salesforce.

The following are the classes in the `Sfdc_Checkout` namespace.

**IN THIS SECTION:**

- **AsyncCartProcessor Interface**
  Use this interface to implement asynchronous integrations in B2B Commerce.

- **B2BCheckoutController Class**
  Communicate with simple checkout Apex methods to work with data related to B2B Commerce checkout.

- **IntegrationInfo Class**
  Provides the values that B2B Commerce Checkout uses to map requests to responses, necessary metadata, and context.

- **IntegrationStatus Class**
  Supports synchronous execution of Apex integrations for B2B Commerce. The implementation must return the status of the execution.
IntegrationStatus.Status Enum
The IntegrationStatus.Status enum describes the status of the current integration.

AsyncCartProcessor Interface
Use this interface to implement asynchronous integrations in B2B Commerce.

Namespace
Sfdc_Checkout

IN THIS SECTION:
AsyncCartProcessor Methods
AsyncCartProcessor Example Implementation

AsyncCartProcessor Methods
The following are methods for AsyncCartProcessor.

IN THIS SECTION:
startCartProcessAsync(integrationInfo, cartId)
The startCartProcessAsync method is called asynchronously by the integration framework. Calling this method begins cart processing for Commerce checkout.

startCartProcessAsync(integrationInfo, cartId)
The startCartProcessAsync method is called asynchronously by the integration framework. Calling this method begins cart processing for Commerce checkout.

Signature
public sfdc_checkout.IntegrationStatus startCartProcessAsync(sfdc_checkout.IntegrationInfo integrationInfo, Id cartId)

Parameters
integrationInfo
Type: IntegrationInfo
Provides values that B2B Commerce checkout APIs use to map requests to responses, necessary metadata, and context.
cartId
Type: Id
ID of the WebCart object.

Return Value
Type: IntegrationStatus
Status of the current integration. Possible values are SUCCESS and FAILED.

AsyncCartProcessor Example Implementation

This is an example implementation of the sfdc_checkout.AsyncCartProcessor interface.

```java
global interface checkout_AsyncCartProcessor {
    //Integration for async processing
    IntegrationStatus startCartProcessAsync(
        IntegrationInfo integrationInfo,
        Id cartId);
}
```

AsyncCartProcessor is a base interface. There are four interfaces that extend it, including CartInventoryValidation, CartPriceCalculations, CartShippingCharges, and CartTaxCalculations. For more information about these interfaces, including code examples and test classes, see Checkout Integrations.

B2BCheckoutController Class

Communicate with simple checkout Apex methods to work with data related to B2B Commerce checkout.

Namespace

sfdc_checkout

Usage

You must specify the sfdc_checkout namespace when creating an instance of this class.

IN THIS SECTION:

B2BCheckoutController Methods

B2BCheckoutController Methods

The following are methods for B2BCheckoutController.

IN THIS SECTION:

licenseCompliance(cartId, orderId)

If you implement your own cart-to-order process without invoking the Cart to Order flow core action, you must invoke this method to correctly track your orders for GMV (Gross Merchandise Value) recognition.

```java
licenseCompliance(cartId, orderId)
```

If you implement your own cart-to-order process without invoking the Cart to Order flow core action, you must invoke this method to correctly track your orders for GMV (Gross Merchandise Value) recognition.

licenseCompliance(cartId, orderId)
Signature

```java
public static void licenseCompliance(String cartId, String orderId)
```

Parameters

cartId
  Type: `String`
  The `cartId` of a web cart from which an order is created.

orderId
  Type: `String`
  The `orderId` of the order you created from the cart.

Return Value

Type: `Void`

IntegrationInfo Class

Provides the values that B2B Commerce Checkout uses to map requests to responses, necessary metadata, and context.

Namespace

`sfdc_checkout` on page 2789

Usage

This class provides information about a B2B Commerce integration. An instance of this class is passed as a parameter into the integration interface.

IN THIS SECTION:

IntegrationInfo Properties

IntegrationInfo Properties

The following are properties for `IntegrationInfo`.

IN THIS SECTION:

- `integrationId`
  The unique ID of a B2B Commerce integration.

- `jobId`
  The ID of the job, specific to the Salesforce Background Operation framework.

- `siteLanguage`
  Site language to be used by third party services.
**integrationId**
The unique ID of a B2B Commerce integration.

**Signature**
```java
public String integrationId {get; set;}
```

**Property Value**
Type: String

**jobId**
The ID of the job, specific to the Salesforce Background Operation framework.

**Signature**
```java
public String jobId {get; set;}
```

**Property Value**
Type: String

**siteLanguage**
Site language to be used by third party services.

**Signature**
```java
public String siteLanguage {get; set;}
```

**Property Value**
Type: String

**IntegrationStatus Class**
Supports synchronous execution of Apex integrations for B2B Commerce. The implementation must return the status of the execution.

**Namespace**
sfdc_checkout

**Usage**
You must specify the sfdc_checkout namespace when creating an instance of this class.

IN THIS SECTION:
  IntegrationStatus Properties
IntegrationStatus Properties
The following are properties for IntegrationStatus.

IN THIS SECTION:

status
Indicates the status of the integration process and whether or not it completed successfully.

status
Indicates the status of the integration process and whether or not it completed successfully.

Signature

public sfdc_checkout.IntegrationStatus.Status status {get; set;}

Property Value
Type: sfdc_checkout.IntegrationStatus.Status on page 2794

IntegrationStatus.Status Enum
The IntegrationStatus.Status enum describes the status of the current integration.

Enum Values
The following are the values of the sfdc_checkout.IntegrationStatus.Status enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FAILED</td>
<td>Indicates transient, unknown error, managed by the implementor. The buyer can retry this action.</td>
</tr>
<tr>
<td>SUCCESS</td>
<td>Indicates the integration executed successfully.</td>
</tr>
</tbody>
</table>

sfdc_surveys Namespace
The sfdc_surveys namespace provides an interface for shortening survey invitations.

The following are the classes in the sfdc_surveys namespace.

IN THIS SECTION:

SurveyInvitationLinkShortener Interface
Use this interface to provide a class factory that Salesforce can call to create instances of your custom SurveyInvitationLinkShortener.

Example Implementation to Associate SurveySubjects with SurveyInvitation and SurveyResponses
If no survey responses are populated, create a custom code to associate SurveySubjects with SurveyInvitation and SurveyResponses.
SurveyInvitationLinkShortener Interface

Use this interface to provide a class factory that Salesforce can call to create instances of your custom SurveyInvitationLinkShortener.

Namespace

sfdc_surveys

Usage

Implement an instance of the SurveyInvitationLinkShortener interface to shorten the survey invitation that can be distributed as short URLs over customer engaged channels, such as SMS, WhatsApp, or Facebook Messenger.

Special access rules

To implement this interface, you must have the Salesforce Feedback Management license enabled in your Salesforce organization.

IN THIS SECTION:

SurveyInvitationLinkShortener Methods

SurveyInvitationLinkShortener Example Implementation

SurveyInvitationLinkShortener Methods

The following are methods for SurveyInvitationLinkShortener.

IN THIS SECTION:

getShortenedURL(var1)

Returns a shortened URL for a given survey invitation.

getShortenedURL(var1)

Returns a shortened URL for a given survey invitation.

IN THIS SECTION:

getShortenedURL(var1)

Returns a shortened URL for a given survey invitation.

getShortenedURL(var1)

Returns a shortened URL for a given survey invitation.

Signature

public String getShortenedURL(String var1)

Parameters

var1

Type: String

Return Value

Type: String
SurveyInvitationLinkShortener Example Implementation

This is an example implementation of the sfdc_surveys.SurveyInvitationLinkShortener interface.
This sample code uses Named Credentials for authentication. For more information on Named Credentials, see Named Credentials as Callout Endpoints.

```java
public class SurveyInvitationLinkShortenerImpl implements sfdc_surveys.SurveyInvitationLinkShortener {
    public String getShortenedURL(String invitationURL) {
        return shortenUrlUsingBitlyService(invitationURL);
    }

    public String shortenUrlUsingBitlyService(String invitationURL) {
        HttpRequest request = new HttpRequest();
        request.setEndpoint('callout:bitly/v4/shorten');
        request.setMethod('POST');
        request.setHeader('Authorization', 'Bearer {!$Credential.Password}');
        request.setHeader('Accept', 'application/json');
        request.setHeader('Content-Type', 'application/json');
        request.setBody(JSON.serialize(new Map<String, Object>{{'group_guid' => '{!$Credential.UserName}', 'long_url' => invitationURL}}));

        Http http = new Http();
        HttpResponse res = http.send(request);

        Object result = JSON.deserializeUntyped(res.getBody());
        if (result instanceof Map<String, Object>) {
            Map<String, Object> resultMap = (Map<String, Object>) result;
            Object shortenedLinkVal = resultMap.get('link');
            if (shortenedLinkVal != null && shortenedLinkVal instanceof String) {
                return (String) shortenedLinkVal;
            }
        }
        return invitationURL;
    }
}
```

Example Implementation to Associate SurveySubjects with SurveyInvitation and SurveyResponses

If no survey responses are populated, create a custom code to associate SurveySubjects with SurveyInvitation and SurveyResponses.

This example shows how to associate SurveySubjects with SurveyInvitation and SurveyResponses.

```java
public class CreateEntriesInSurveyInvitationRespRL {
    // Utility to create SurveyInvitation and SurveySubject record
    public static void addEntry(String associatedRecordId, String surveyId, String participantId) {
        String invitationId = createSurveyInvitation(surveyId, participantId);
        createSurveySubject(invitationId, associatedRecordId);
    }
}
```
private static String createSurveyInvitation(String surveyId, String participantId) {
    SurveyInvitation surveyInv = new SurveyInvitation();
    surveyInv.Name = 'SurveyInvitationForCase'; // add your survey invitation name here
    surveyInv.ParticipantId = participantId;
    surveyInv.CommunityId = '0DBRM0000004n4y'; // add your community id here
    surveyInv.OptionsAllowGuestUserResponse = true;
    surveyInv.SurveyId = surveyId;

    // Insert the SurveyInvitation Record
    insert surveyInv;

    return surveyInv.Id;
}

private static void createSurveySubject(String invitationId, String associatedRecordId) {
    SurveySubject subj = new SurveySubject();
    subj.Name = 'Sur_Subject_for_invitation';
    subj.ParentId = invitationId; // similarly you can use survey response id to associate survey subject to a response record.
    subj.SubjectId = associatedRecordId;

    // Insert the SurveySubject Record
    insert subj;
}

trigger SurveyResponseForCaseTrigger on SurveyResponse (after insert) {
    System.debug('Inside Survey response trigger ');
    for(SurveyResponse sr: Trigger.New) {
        SurveySubject subj = new SurveySubject();
        subj.Name = 'Sur_Subject_for_response';
        subj.ParentId = sr.id; // Associating survey response id
        subj.SubjectId = associatedRecordId;

        // Get the associatedRecordId recordId (like Case, Opportunity etc) using the SurveyInvitation Id and
        // assigning it to SubjectId, assuming we inserted SurveySubject record for the associated invitation
        // using the previous code
        List<SurveySubject> SurSubj=[select subjectid from SurveySubject where parentid =

        List<SurveySubject> SurSubj=[select subjectid from SurveySubject where parentid =

        List<SurveySubject> SurSubj=[select subjectid from SurveySubject where parentid =
for (SurveySubject sub: SurSubj) {
    String ids = String.valueOf(sub.subjectid).substring(0, 3);
    if ('500'.equals(ids)) {
        subj.SubjectId = sub.subjectid;
        // Insert the SurveySubject Record
        insert subj;
        break;
    }
}

Site Namespace

The Site namespace provides an interface for rewriting Sites URLs.
The following is the interface in the Site namespace.

IN THIS SECTION:

- UrlRewriter Interface
  Enables rewriting Sites URLs.
- Site Exceptions
  The Site namespace contains an exception class.

UrlRewriter Interface

Enables rewriting Sites URLs.

Namespace

Site

Usage

Sites provides built-in logic that helps you display user-friendly URLs and links to site visitors. Create rules to rewrite URL requests typed into the address bar, launched from bookmarks, or linked from external websites. You can also create rules to rewrite the URLs for links within site pages. URL rewriting not only makes URLs more descriptive and intuitive for users, it allows search engines to better index your site pages.

For example, let’s say that you have a blog site. Without URL rewriting, a blog entry's URL might look like this: https://myblog.my.salesforce-sites.com/posts?id=003D000000QOPcN

To rewrite URLs for a site, create an Apex class that maps the original URLs to user-friendly URLs, and then add the Apex class to your site.
UrlRewriter Methods

The following are methods for UrlRewriter. All are instance methods.

IN THIS SECTION:
  generateUrlFor(salesforceUrls)
  Maps a list of Salesforce URLs to a list of user-friendly URLs.
  mapRequestUrl(userFriendlyUrl)
  Maps a user-friendly URL to a Salesforce URL.

generateUrlFor(salesforceUrls)
Maps a list of Salesforce URLs to a list of user-friendly URLs.

Signature
public System.PageReference[] generateUrlFor(System.PageReference[] salesforceUrls)

Parameters
salesforceUrls
  Type: System.PageReference[]

Return Value
Type: System.PageReference[]

Usage
You can use List<PageReference> instead of PageReference [], if you prefer.

⚠️ Important: The size and order of the input list of Salesforce URLs must exactly correspond to the size and order of the generated list of user-friendly URLs. The generateUrlFor method maps input URLs to output URLs based on the order in the lists.

mapRequestUrl(userFriendlyUrl)
Maps a user-friendly URL to a Salesforce URL.

Signature
public System.PageReference mapRequestUrl(System.PageReference userFriendlyUrl)

Parameters
userFriendlyUrl
  Type: System.PageReference

Return Value
Type: System.PageReference
Site Exceptions

The Site namespace contains an exception class. All exception classes support built-in methods for returning the error message and exception type. See Exception Class and Built-In Exceptions.

The Site namespace contains this exception:

<table>
<thead>
<tr>
<th>Exception</th>
<th>Description</th>
<th>Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>Site.ExternalUserCreateException</td>
<td>Unable to create external user</td>
<td>Use the String getMessage() to get the error message and write it to debug log.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Use List&lt;String&gt; getDisplayMessages() to get a list of errors displayed to the end user.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This exception can’t be subclassed or thrown in code.</td>
</tr>
</tbody>
</table>

Support Namespace

The Support namespace provides an interface used for Case Feed. The following is the interface in the Support namespace.

IN THIS SECTION:

EmailTemplateSelector Interface
The Support.EmailTemplateSelector interface enables providing default email templates in Case Feed. With default email templates, specified email templates are preloaded for cases based on criteria such as case origin or subject.

MilestoneTriggerTimeCalculator Interface
The Support.MilestoneTriggerTimeCalculator interface calculates the time trigger for a milestone.

EmailTemplateSelector Interface

The Support.EmailTemplateSelector interface enables providing default email templates in Case Feed. With default email templates, specified email templates are preloaded for cases based on criteria such as case origin or subject.

Namespace

Support
To specify default templates, you must create a class that implements Support.EmailTemplateSelector.

When you implement this interface, provide an empty parameterless constructor.

IN THIS SECTION:

EmailTemplateSelector Methods
EmailTemplateSelector Example Implementation
EmailTemplateSelector Methods

The following are methods for EmailTemplateSelector.

IN THIS SECTION:

getDefaultTemplateId(caseId)
Returns the ID of the email template to preload for the case currently being viewed in the case feed using the specified case ID.

**getDefaultTemplateId(caseId)**

Returns the ID of the email template to preload for the case currently being viewed in the case feed using the specified case ID.

**Signature**

```java
public ID getDefaultTemplateId(ID caseId)
```

**Parameters**

caseId
Type: ID

**Return Value**
Type: ID

EmailTemplateSelector Example Implementation

This is an example implementation of the Support.EmailTemplateSelector interface.

The `getDefaultEmailTemplateId` method implementation retrieves the subject and description of the case corresponding to the specified case ID. Next, it selects an email template based on the case subject and returns the email template ID.

```java
global class MyCaseTemplateChooser implements Support.EmailTemplateSelector {
    // Empty constructor
    global MyCaseTemplateChooser() { }

    // The main interface method
    global ID getDefaultEmailTemplateId(ID caseId) {
        // Select the case we're interested in, choosing any fields that are relevant to our decision
        Case c = [SELECT Subject, Description FROM Case WHERE Id=:caseId];

        EmailTemplate et;
        if (c.subject.contains('LX-1150')) {
            et = [SELECT id FROM EmailTemplate WHERE DeveloperName = 'LX1150_template'];
        } else if(c.subject.contains('LX-1220')) {
            et = [SELECT id FROM EmailTemplate WHERE DeveloperName = 'LX1220_template'];
        }

        // Return the ID of the template selected
        return et.id;
    }
}```
The following example tests the above code:

```java
@isTest
private class MyCaseTemplateChooserTest {

    static testMethod void testChooseTemplate() {

        MyCaseTemplateChooser chooser = new MyCaseTemplateChooser();

        // Create a simulated case to test with
        Case c = new Case();
        c.Subject = 'I\'m having trouble with my LX-1150';
        Database.insert(c);

        // Make sure the proper template is chosen for this subject
        Id actualTemplateId = chooser.getDefaultEmailTemplateId(c.Id);
        EmailTemplate expectedTemplate =
            [SELECT id FROM EmailTemplate WHERE DeveloperName = 'LX1150_template'];
        Id expectedTemplateId = expectedTemplate.Id;
        System.assertEquals(actualTemplateId, expectedTemplateId);

        // Change the case properties to match a different template
        c.Subject = 'My LX1220 is overheating';
        Database.update(c);

        // Make sure the correct template is chosen in this case
        actualTemplateId = chooser.getDefaultEmailTemplateId(c.Id);
        expectedTemplate =
            [SELECT id FROM EmailTemplate WHERE DeveloperName = 'LX1220_template'];
        expectedTemplateId = expectedTemplate.Id;
        System.assertEquals(actualTemplateId, expectedTemplateId);
    }
}
```

**MilestoneTriggerTimeCalculator Interface**

The `Support.MilestoneTriggerTimeCalculator` interface calculates the time trigger for a milestone.

**Namespace**

`Support`

Implement the `Support.MilestoneTriggerTimeCalculator` interface to calculate a dynamic time trigger for a milestone based on the milestone type, the properties of the case, and case-related objects. To implement the `Support.MilestoneTriggerTimeCalculator` interface, you must first declare a class with the `implements` keyword as follows:

```java
global class Employee implements Support.MilestoneTriggerTimeCalculator {
```
Next, your class must provide an implementation for the following method:

```plaintext
global Integer calculateMilestoneTriggerTime(String caseId, String milestoneTypeId)
```

The implemented method must be declared as `global` or `public`.

IN THIS SECTION:
- MilestoneTriggerTimeCalculator Methods
- MilestoneTriggerTimeCalculator Example Implementation

**MilestoneTriggerTimeCalculator Methods**

The following are instance methods for `MilestoneTriggerTimeCalculator`.

IN THIS SECTION:
- `calculateMilestoneTriggerTime(caseId, milestoneTypeId)`
  - Calculates the milestone trigger time based on the specified case and milestone type and returns the time in minutes.

**calculateMilestoneTriggerTime(caseId, milestoneTypeId)**

Calculates the milestone trigger time based on the specified case and milestone type and returns the time in minutes.

**Syntax**

```plaintext
public Integer calculateMilestoneTriggerTime(String caseId, String milestoneTypeId)
```

**Parameters**

- `caseId`
  - Type: String
  - ID of the case the milestone is applied to.

- `milestoneTypeId`
  - Type: String
  - ID of the milestone type.

**Return Value**

Type: Integer

The calculated trigger time in minutes.

**MilestoneTriggerTimeCalculator Example Implementation**

This sample class demonstrates the implementation of the `Support.MilestoneTriggerTimeCalculator` interface. In this sample, the case's priority and the milestone `m1` determine that the time trigger is 18 minutes.

```plaintext
global class myMilestoneTimeCalculator implements Support.MilestoneTriggerTimeCalculator {

2803
```
global Integer calculateMilestoneTriggerTime(String caseId, String milestoneTypeId) {
    Case c = [SELECT Priority FROM Case WHERE Id=:caseId];
    MilestoneType mt = [SELECT Name FROM MilestoneType WHERE Id=:milestoneTypeId];
    if (c.Priority != null && c.Priority.equals('High')){
        if (mt.Name != null && mt.Name.equals('m1')) { return 7; }
        else { return 5; }
    }
    else {
        return 18;
    }
}

This test class can be used to test the implementation of Support.MilestoneTriggerTimeCalculator.

@isTest
private class MilestoneTimeCalculatorTest {
    static testMethod void testMilestoneTimeCalculator() {
        // Select an existing milestone type to test with
        MilestoneType[] mtLst = [SELECT Id, Name FROM MilestoneType LIMIT 1];
        if(mtLst.size() == 0) { return; }
        MilestoneType mt = mtLst[0];

        // Create case data.
        // Typically, the milestone type is related to the case,
        // but for simplicity, the case is created separately for this test.
        Case c = new Case(priority = 'High');
        insert c;

        myMilestoneTimeCalculator calculator = new myMilestoneTimeCalculator();
        Integer actualTriggerTime = calculator.calculateMilestoneTriggerTime(c.Id, mt.Id);

        if(mt.name != null && mt.Name.equals('m1')) {
            System.assertEquals(actualTriggerTime, 7);
        }
        else {
            System.assertEquals(actualTriggerTime, 5);
        }

        c.priority = 'Low';
        update c;
        actualTriggerTime = calculator.calculateMilestoneTriggerTime(c.Id, mt.Id);
        System.assertEquals(actualTriggerTime, 18);
    }
}

System Namespace

The System namespace provides classes and methods for core Apex functionality.

The following are the classes in the System namespace.
IN THIS SECTION:

**AccessLevel Class**
Defines the different modes, such as system or user mode, that Apex database operations execute in.

**AccessType Enum**
Specifies the access check type for the fields of an sObject.

**Address Class**
Contains methods for accessing the component fields of address compound fields.

**Answers Class**
Represents zone answers.

**ApexPages Class**
Use `ApexPages` to add and check for messages associated with the current page, as well as to reference the current page.

**Approval Class**
Contains methods for processing approval requests and setting approval-process locks and unlocks on records.

**Assert Class**
Contains methods to assert various conditions with test methods, such as whether two values are the same, a condition is true, or a variable is null.

**AsyncInfo Class**
Provides methods to get the current stack depth, maximum stack depth, and the minimum queueable delay for Queueable transactions, and to determine if maximum stack depth is set.

**AsyncOptions Class**
Contains maximum stack depths for queueable transactions and the minimum queueable delay in minutes. Passed as parameter to the `System.enqueueJob()` method to define a unique queueable job signature, the maximum stack depth for queueable transactions and the minimum queueable delay in minutes.

**Blob Class**
Contains methods for the Blob primitive data type.

**Boolean Class**
Contains methods for the Boolean primitive data type.

**BusinessHours Class**
Use the `BusinessHours` methods to set the business hours at which your customer support team operates.

**Callable Interface**
Enables developers to use a common interface to build loosely coupled integrations between Apex classes or triggers, even for code in separate packages. Agreeing upon a common interface enables developers from different companies or different departments to build upon one another’s solutions. Implement this interface to enable the broader community, which might have different solutions than the ones you had in mind, to extend your code’s functionality.

**Cases Class**
Use the `Cases` class to interact with case records.

**Collator Class**
Contains methods to get locale-specific instances that can be used for comparisons and sorting. Use the `getInstance()` method to obtain the Collator instance for a given locale and pass the Collator as the Comparator parameter to the `list.sort()` method.
Comparable Interface
Adds sorting support for Lists that contain non-primitive types, that is, Lists of user-defined types. Your implementation must explicitly handle null inputs in the `compareTo()` method to avoid a null pointer exception.

Comparator Interface
Implement different sort orders with the Comparator interface’s `compare()` method, and pass the Comparator as a parameter to `List.sort()`. Your implementation must explicitly handle null inputs in the `compare()` method to avoid a null pointer exception.

Continuation Class
Use the `Continuation` class to make callouts asynchronously to a SOAP or REST Web service.

Cookie Class
The `Cookie` class lets you access cookies for your Salesforce site using Apex.

Crypto Class
Provides methods for creating digests, message authentication codes, and signatures, as well as encrypting and decrypting information.

Custom Metadata Type Methods
Custom metadata types are customizable, deployable, packageable, and upgradeable application metadata. All custom metadata is exposed in the application cache, which allows access without repeated queries to the database. The metadata is then available for formula fields, validation rules, flows, Apex, and SOAP API. All methods are static.

Custom Settings Methods
Custom settings are similar to custom objects and enable application developers to create custom sets of data, as well as create and associate custom data for an organization, profile, or specific user. All custom settings data is exposed in the application cache, which enables efficient access without the cost of repeated queries to the database. This data is then available for formula fields, validation rules, flows, Apex, and the SOAP API.

Database Class
Contains methods for creating and manipulating data.

Date Class
Contains methods for the Date primitive data type.

Datetime Class
Contains methods for the Datetime primitive data type.

Decimal Class
Contains methods for the Decimal primitive data type.

Domain Class
Represents an existing domain hosted by Salesforce that serves the org or its content. Contains methods to obtain information about these domains, such as the domain type, My Domain name, and sandbox name.

DomainCreator Class
Use the `DomainCreator` class to return a hostname specific to the org. For example, get the org’s Visualforce hostname. Values are returned as a hostname, such as `MyDomainName.lightning.force.com`.

DomainParser Class
Use the `DomainParser` class to parse a domain that Salesforce hosts for the org and extract information about the domain.

DomainType Enum
Specifies the domain type for a `System.Domain`.

Double Class
Contains methods for the Double primitive data type.
EmailMessages Class
Use the methods in the EmailMessages class to interact with emails and email threading.

EncodingUtil Class
Use the methods in the EncodingUtil class to encode and decode URL strings, and convert strings to hexadecimal format.

Enum Methods
An enum is an abstract data type with values that each take on exactly one of a finite set of identifiers that you specify. Apex provides built-in enums, such as LoggingLevel, and you can define your own enum.

EventBus Class
Contains methods for publishing platform events.

Exception Class and Built-In Exceptions
An exception denotes an error that disrupts the normal flow of code execution. You can use Apex built-in exceptions or create custom exceptions. All exceptions have common methods.

FlexQueue Class
Contains methods that reorder batch jobs in the Apex flex queue.

FeatureManagement Class
Use the methods in the System.FeatureManagement class to check and modify the values of feature parameters, and to show or hide custom objects and custom permissions in your subscribers' orgs.

Formula Class
Contains the recalculateFormulas method that updates (recalculates) all formula fields on the input SObjects.

FormulaRecalcFieldError Class
The return type of the FormulaRecalcResult.getErrors method.

FormulaRecalcResult Class
The return type of the Formula.recalculateFormulas method.

Http Class
Use the Http class to initiate an HTTP request and response.

HttpCalloutMock Interface
Enables sending fake responses when testing HTTP callouts.

HttpRequest Class
Use the HttpRequest class to programmatically create HTTP requests like GET, POST, PATCH, PUT, and DELETE.

HttpResponse Class
Use the HttpResponse class to handle the HTTP response returned by the Http class.

Id Class
Contains methods for the ID primitive data type.

Ideas Class
Represents zone ideas.

InstallHandler Interface
Enables custom code to run after a managed package installation or upgrade.

Integer Class
Contains methods for the Integer primitive data type.
JSON Class
Contains methods for serializing Apex objects into JSON format and deserializing JSON content that was serialized using the `serialize` method in this class.

JSONGenerator Class
Contains methods used to serialize objects into JSON content using the standard JSON encoding.

JSONParser Class
Represents a parser for JSON-encoded content.

JSONToken Enum
Contains all token values used for parsing JSON content.

Label Class
Provides methods to retrieve a custom label or to check if translation exists for a label in a specific language and namespace. Label names are dynamically resolved at run time, overriding the user’s current language if a translation exists for the requested language. You can’t access labels that are protected in a different namespace.

Limits Class
Contains methods that return limit information for specific resources.

List Class
Contains methods for the List collection type.

Location Class
Contains methods for accessing the component fields of geolocation compound fields.

LoggingLevel Enum
Specifies the logging level for the `System.debug` method.

Long Class
Contains methods for the Long primitive data type.

Map Class
Contains methods for the Map collection type.

Matcher Class
Matchers use Patterns to perform match operations on a character string.

Math Class
Contains methods for mathematical operations.

Messaging Class
Contains messaging methods used when sending a single or mass email.

MultiStaticResourceCalloutMock Class
Utility class used to specify a fake response using multiple resources for testing HTTP callouts.

Network Class
Manage Experience Cloud sites.

OrgLimit Class
Contains methods that provide the name, maximum value, and current value of an org limit.

OrgLimits Class
Contains methods that provide a list or map of all OrgLimit instances for Salesforce your org, such as SOAP API requests, Bulk API requests, and Streaming API limits.
**PageReference Class**
A PageReference is a reference to an instantiation of a page. Among other attributes, PageReferences consist of a URL and a set of query parameter names and values.

**Packaging Class**
Contains a method for obtaining information about managed and unlocked packages.

**Pattern Class**
Represents a compiled representation of a regular expression.

**Queueable Interface**
Enables the asynchronous execution of Apex jobs that can be monitored.

**QueueableContext Interface**
Represents the parameter type of the `execute()` method in a class that implements the `Queueable` interface and contains the job ID. This interface is implemented internally by Apex.

**QueueableDuplicateSignature Class**
Used in the `AsyncOptions` class to store the queueable job signature in the `DuplicateSignature` property.

**QueueableDuplicateSignature.Builder Class**
Build a unique signature for your queueable job using this inner builder class. The `build()` class method builds a `QueueableDuplicateSignature` object, with input from the `addId()`, `addInteger()`, and `addString()` methods. Use the `DuplicateSignature` property in the `AsyncOptions` class to store the queueable job signature. Enqueue your job by using the `System.enqueueJob()` with the `AsyncOptions` parameter.

**QuickAction Class**
Use Apex to request and process actions on objects that allow custom fields, on objects that appear in a Chatter feed, or on objects that are available globally.

**Quiddity Enum**
Specifies a Quiddity value used by the methods in the `System.Request` class.

**RemoteObjectController**
Use `RemoteObjectController` to access the standard Visualforce Remote Objects operations in your Remote Objects override methods.

**Request Class**
Contains methods to obtain the request ID and Quiddity value of the current Salesforce request.

**ResetPasswordResult Class**
Represents the result of a password reset.

**RestContext Class**
Contains the `RestRequest` and `RestResponse` objects.

**RestRequest Class**
Use the `System.RestRequest` class to access and pass request data in a RESTful Apex method.

**RestResponse Class**
Represents an object used to pass data from an Apex RESTful Web service method to an HTTP response.

**SandboxPostCopy Interface**
To make your sandbox environment business ready, automate data manipulation or business logic tasks. Extend this interface and add methods to perform post-copy tasks, then specify the class during sandbox creation.

**Schedulable Interface**
The class that implements this interface can be scheduled to run at different intervals.
**SchedulableContext Interface**

Represents the parameter type of a method in a class that implements the Schedulable interface and contains the scheduled job ID. This interface is implemented internally by Apex.

**Schema Class**

Contains methods for obtaining schema describe information.

**Search Class**

Use the methods of the Search class to perform dynamic SOSL queries.

**Security Class**

Contains methods to securely implement Apex applications.

**SelectOption Class**

A SelectOption object specifies one of the possible values for a Visualforce selectCheckboxes, selectList, or selectRadio component.

**Set Class**

Represents a collection of unique elements with no duplicate values.

**Site Class**

Use the Site Class to manage your sites. Change, reset, validate, and check the expiration of passwords. Create site users, person accounts, and portal users. Get the admin email and ID. Get various URLs, the path prefix, the ID, the template, and the type of the site. Log in to the site.

**SObject Class**

Contains methods for the sObject data type.

**SObjectAccessDecision Class**

Contains the results of a call to the Security.stripInaccessible method and methods to retrieve those results.

**StaticResourceCalloutMock Class**

Utility class used to specify a fake response for testing HTTP callouts.

**String Class**

Contains methods for the String primitive data type.

**StubProvider Interface**

StubProvider is a callback interface that you can use as part of the Apex stub API to implement a mocking framework. Use this interface with the Test.createStub() method to create stubbed Apex objects for testing.

**System Class**

Contains methods for system operations, such as writing debug messages and scheduling jobs.

**Test Class**

Contains methods related to Apex tests.

**Time Class**

Contains methods for the Time primitive data type.

**TimeZone Class**

Represents a time zone. Contains methods for creating a new time zone and obtaining time zone properties, such as the time zone ID, offset, and display name.

**Trigger Class**

Use the Trigger class to access run-time context information in a trigger, such as the type of trigger or the list of sObject records that the trigger operates on.
TriggerOperation Enum
System.TriggerOperation enum values are associated with trigger events.

Type Class
Contains methods for getting the Apex type that corresponds to an Apex class and for instantiating new types.

UninstallHandler Interface
Enables custom code to run after a managed package is uninstalled.

URL Class
Represents a uniform resource locator (URL) and provides access to parts of the URL. Enables access to the base URL used to access your Salesforce org.

UserInfo Class
Contains methods for obtaining information about the context user.

UserManagement Class
Contains methods to manage end users, for example, to register their verification methods, verify their identity, or remove their personal information.

Version Class
Use the Version methods to get the version of a first-generation managed package, and to compare package versions.

WebServiceCallout Class
Enables making callouts to SOAP operations on an external Web service. This class is used in the Apex stub class that is auto-generated from a WSDL.

WebServiceMock Interface
Enables sending fake responses when testing Web service callouts of a class auto-generated from a WSDL.

XmlStreamReader Class
The XmlStreamReader class provides methods for forward, read-only access to XML data. You can pull data from XML or skip unwanted events. You can parse nested XML content that’s up to 50 nodes deep.

XmlStreamWriter Class
The XmlStreamWriter class provides methods for writing XML data.

AccessLevel Class
Defines the different modes, such as system or user mode, that Apex database operations execute in.

Namespace
System

Usage
By default, Apex code runs in system mode, which means that it runs with substantially elevated permissions over the user running the code. In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the current user’s object permissions, field-level security, and sharing rules are enforced. Many of the DML methods of the System.Database and System.Search classes include an accessLevel parameter to specify the execution mode.
Example

If the user running this Apex code doesn’t have write access to the Account object, the `Database.insert()` method returns an error.

```java
List<Account> toInsert = new List<Account>{new Account(Name = 'Exciting New Account')};
List<Database.SaveResult> sr = Database.insert(toInsert, AccessLevel.USER_MODE);
```

In contrast, this example shows the method running in system mode. The success of the insert doesn’t depend on whether the user running the Apex code has create access to the Account object.

```java
List<Account> toInsert = new List<Account>{new Account(Name = 'Exciting New Account')};
List<Database.SaveResult> sr = Database.insert(toInsert, AccessLevel.SYSTEM_MODE);
```

IN THIS SECTION:

- AccessLevel Methods
- AccessLevel Properties

AccessLevel Methods

The following are methods for `AccessLevel`.

IN THIS SECTION:

- `withPermissionSetId(permissionSetId)` (Developer Preview)

  Supports database and search operations to be run with permissions specified in a permission set. Apex enforces field-level security (FLS) and object permissions as per the specified permission set, in addition to the running user’s permissions.

`withPermissionSetId(permissionSetId)` (Developer Preview)

Supports database and search operations to be run with permissions specified in a permission set. Apex enforces field-level security (FLS) and object permissions as per the specified permission set, in addition to the running user’s permissions.

**Note:** Feature is available as a developer preview. Feature isn’t generally available unless or until Salesforce announces its general availability in documentation or in press releases or public statements. All commands, parameters, and other features are subject to change or deprecation at any time, with or without notice. Don’t implement functionality developed with these commands or tools in a production environment. You can provide feedback and suggestions for the “Permission Sets with User Mode” feature in the Trailblazer Community.

This feature is available in scratch orgs where the `ApexUserModeWithPermset` feature is enabled. If the feature isn’t enabled, Apex code with this feature can be compiled but not executed.

**Signature**

```java
public System.AccessLevel withPermissionSetId(String permissionSetId)
```
Parameters

permissionSetId
Type: String

Permissions in the specified permission set are enforced while running user-mode DML operations, in addition to the running user’s permissions.

Return Value

Type: Access Level Class

Example: This example runs the AccessLevel.withPermissionSetId() method with the specified permission set and inserts a custom object.

```java
@isTest
public with sharing class ElevateUserModeOperations_Test {
    @isTest
    static void objectCreatePermViaPermissionSet() {
        Profile p = [SELECT Id FROM Profile WHERE Name='Minimum Access - Salesforce'];

        User u = new User(Alias = 'standt', Email='standarduser@testorg.com',
                          EmailEncodingKey='UTF-8', LastName='Testing', LanguageLocaleKey='en_US',
                          LocaleSidKey='en_US', ProfileId = p.Id,
                          TimeZoneSidKey='America/Los_Angeles',
                          UserName='standarduser' + DateTime.now().getTime() + '@testorg.com');

        System.runAs(u) {
            try {
                Database.insert(new Account(name='foo'), AccessLevel.User_mode);
                Assert.fail();
            } catch (SecurityException ex) {
                Assert.isTrue(ex.getMessage().contains('Account'));
            }

            Id permissionSetId = [Select Id from PermissionSet
                                  where Name = 'AllowCreateToAccount' limit 1].Id;

            Database.insert(new Account(name='foo'),
                            AccessLevel.User_mode.withPermissionSetId(permissionSetId));

            // The elevated access level in not persisted to subsequent operations
            try {
                Database.insert(new Account(name='foo2'), AccessLevel.User_mode);
                Assert.fail();
            } catch (SecurityException ex) {
```
Assert.isTrue(ex.getMessage().contains('Account'));

AccessLevel Properties

The following are properties for AccessLevel.

IN THIS SECTION:

SYSTEM_MODE
Execution mode in which the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords.

USER_MODE
Execution mode in which the object permissions, field-level security, and sharing rules of the current user are enforced.

SYSTEM_MODE

Execution mode in which the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords.

Signature

public System.AccessLevel SYSTEM_MODE {get;}

Property Value
Type: System.AccessLevel

USER_MODE

Execution mode in which the object permissions, field-level security, and sharing rules of the current user are enforced.

Signature

public System.AccessLevel USER_MODE {get;}

Property Value
Type: System.AccessLevel

AccessType Enum

Specifies the access check type for the fields of an sObject.
Usage

Use these enum values for the `accessCheckType` parameter of the `stripInaccessible` method.

Enum Values

The following are the values of the `System.AccessType` enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CREATABLE</td>
<td>Check the fields of an sObject for create access.</td>
</tr>
<tr>
<td>READABLE</td>
<td>Check the fields of an sObject for read access.</td>
</tr>
<tr>
<td>UPDATABLE</td>
<td>Check the fields of an sObject for update access.</td>
</tr>
<tr>
<td>UPSERTABLE</td>
<td>Check the fields of an sObject for both insert and update access.</td>
</tr>
</tbody>
</table>

Address Class

Contains methods for accessing the component fields of address compound fields.

Namespace

System

Usage

Each of these methods is also equivalent to a read-only property. For each getter method, you can access the property using dot notation. For example, `myAddress.getCity()` is equivalent to `myAddress.city`.

You can’t use dot notation to access compound fields’ subfields directly on the parent field. Instead, assign the parent field to a variable of type `Address`, and then access its components. For example, to access the `City` field in `myAccount.BillingAddress`, do the following:

```java
Address addr = myAccount.BillingAddress;
String acctCity = addr.City;
```

⚠️ **Important:** “Address” in Salesforce can also refer to the Address standard object. When referencing the Address object in your Apex code, always use `Schema.Address` instead of `Address` to prevent confusion with the standard Address compound field. If referencing both the Address object and the Address standard field in the same snippet, you can differentiate between the two by using `System.Address` for the field and `Schema.Address` for the object.

Example

```java
// Select and access Address fields.
// Call the getDistance() method in different ways.
Account[] records = [SELECT id, BillingAddress FROM Account LIMIT 10];
for(Account acct : records) {
    Address addr = acct.BillingAddress;
```
Double lat = addr.latitude;
Double lon = addr.longitude;
Location loc1 = Location.newInstance(30.1944,-97.6682);
Double apexDist1 = addr.getDistance(loc1, 'mi');
Double apexDist2 = loc1.getDistance(addr, 'mi');
System.assertEquals(apexDist1, apexDist2);
Double apexDist3 = Location.getDistance(addr, loc1, 'mi');
System.assertEquals(apexDist2, apexDist3);
}

IN THIS SECTION:
Address Methods

Address Methods

The following are methods for Address.

IN THIS SECTION:
getCity()  
Returns the city field of this address.

getCountry() 
Returns the text-only country/territory name component of this address.

getCountryCode() 
Returns the country/territory code of this address if state and country/territory picklists are enabled in your organization. Otherwise, returns null.

getAddress(toLocation, unit)  
Returns the distance from this location to the specified location using the specified unit.

getGeocodeAccuracy()  
When using geolocation data for a given address, this method gives you relative location information based on latitude and longitude values. For example, you can find out if the latitude and longitude values point to the middle of the street, instead of the exact address.

getLatitude()  
Returns the latitude field of this address.

getLongitude()  
Returns the longitude field of this address.

getPostalCode()  
Returns the postal code of this address.

getState()  
Returns the text-only state name component of this address.

getStateCode()  
Returns the state code of this address if state and country/territory picklists are enabled in your organization. Otherwise, returns null.
**getStreet()**
Returns the street field of this address.

**getCity()**
Returns the city field of this address.

**Signature**
```
public String getCity()
```

**Return Value**
Type: String

**getCountry()**
Returns the text-only country/territory name component of this address.

**Signature**
```
public String getCountry()
```

**Return Value**
Type: String

**getCountryCode()**
Returns the country/territory code of this address if state and country/territory picklists are enabled in your organization. Otherwise, returns null.

**Signature**
```
public String getCountryCode()
```

**Return Value**
Type: String

**getDistance(toLocation, unit)**
Returns the distance from this location to the specified location using the specified unit.

**Signature**
```
public Double getDistance(Location toLocation, String unit)
```
Parameters

toLocation
  Type: Location
  The Location to which you want to calculate the distance from the current Location.

unit
  Type: String
  The distance unit you want to use: mi or km.

Return Value
  Type: Double

getGeocodeAccuracy()
When using geolocation data for a given address, this method gives you relative location information based on latitude and longitude values. For example, you can find out if the latitude and longitude values point to the middle of the street, instead of the exact address.

Signature
  public String getGeocodeAccuracy()

Return Value
  Type: String
  The getGeocodeAccuracy() return value tells you more about the location at a latitude and longitude for a given address. For example, Zip means the latitude and longitude point to the center of the zip code area, in case a match for an exact street address can’t be found.

<table>
<thead>
<tr>
<th>Accuracy Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
<td>In the same building</td>
</tr>
<tr>
<td>NearAddress</td>
<td>Near the address</td>
</tr>
<tr>
<td>Block</td>
<td>Midway point of the block</td>
</tr>
<tr>
<td>Street</td>
<td>Midway point of the street</td>
</tr>
<tr>
<td>ExtendedZip</td>
<td>Center of the extended zip code area</td>
</tr>
<tr>
<td>Zip</td>
<td>Center of the zip code area</td>
</tr>
<tr>
<td>Neighborhood</td>
<td>Center of the neighborhood</td>
</tr>
<tr>
<td>City</td>
<td>Center of the city</td>
</tr>
<tr>
<td>County</td>
<td>Center of the county</td>
</tr>
<tr>
<td>State</td>
<td>Center of the state</td>
</tr>
<tr>
<td>Unknown</td>
<td>No match for the address was found</td>
</tr>
</tbody>
</table>
Geocodes are added only for some standard addresses.

- Billing Address on accounts
- Shipping Address on accounts
- Mailing Address on contacts
- Address on leads

Person accounts are not supported.

**Note:** For `getGeocodeAccuracy()` to work, set up and activate the geocode data integration rules for the related address fields.

### getLatitude()

Returns the latitude field of this address.

**Signature**

```java
public Double getLatitude()
```

**Return Value**

Type: **Double**

### getLongitude()

Returns the longitude field of this address.

**Signature**

```java
public Double getLongitude()
```

**Return Value**

Type: **Double**

### getPostalCode()

Returns the postal code of this address.

**Signature**

```java
public String getPostalCode()
```

**Return Value**

Type: **String**

### getState()

Returns the text-only state name component of this address.
Signature
public String getState()

Return Value
Type: String

getStateCode()
Returns the state code of this address if state and country/territory picklists are enabled in your organization. Otherwise, returns null.

Signature
public String getStateCode()

Return Value
Type: String

getStreet()
Returns the street field of this address.

Signature
public String getStreet()

Return Value
Type: String

Answers Class
Represents zone answers.

Namespace
System

Usage
Answers is a feature that enables users to ask questions and have zone members post replies. Members can then vote on the helpfulness of each reply, and the person who asked the question can mark one reply as the best answer.

For more information on answers, see “Answers Overview” in the Salesforce online help.
Example

The following example finds questions in an internal zone that have similar titles as a new question:

```java
public class FindSimilarQuestionController {

    public static void test() {
        // Instantiate a new question
        Question question = new Question();

        // Specify a title for the new question
        question.title = 'How much vacation time do full-time employees get?';

        // Specify the communityID (INTERNAL_COMMUNITY) in which to find similar questions.
        Community community = [ SELECT Id FROM Community WHERE Name = 'INTERNAL_COMMUNITY' ];

        question.communityId = community.id;

        ID[] results = Answers.findSimilar(question);
    }
}
```

The following example marks a reply as the best reply:

```java
ID questionId = [SELECT Id FROM Question WHERE Title = 'Testing setBestReplyId' LIMIT 1].Id;
ID replyId = [SELECT Id FROM Reply WHERE QuestionId = :questionId LIMIT 1].Id;
Answers.setBestReply(questionId,replyId);
```

Answers Methods

The following are methods for Answers. All methods are static.

**findSimilar(yourQuestion)**

Returns a list of similar questions based on the title of the specified question.

**setBestReply(questionId, replyId)**

Sets the specified reply for the specified question as the best reply. Because a question can have multiple replies, setting the best reply helps users quickly identify the reply that contains the most helpful information.

**findSimilar (yourQuestion)**

Returns a list of similar questions based on the title of the specified question.

**Signature**

```java
public static ID[] findSimilar(Question yourQuestion)
```
Parameters

yourQuestion
  Type: Question

Return Value

Type: ID[]

Usage

Each findSimilar call counts against the SOSL statements governor limit allowed for the process.

setBestReply(questionId, replyId)

Sets the specified reply for the specified question as the best reply. Because a question can have multiple replies, setting the best reply helps users quickly identify the reply that contains the most helpful information.

Signature

public static Void setBestReply(String questionId, String replyId)

Parameters

questionId
  Type: String
replyId
  Type: String

Return Value

Type: Void

ApexPages Class

Use ApexPages to add and check for messages associated with the current page, as well as to reference the current page.

Namespace

System

Usage

In addition, ApexPages is used as a namespace for the PageReference Class and the Message Class.

ApexPages Methods

The following are methods for ApexPages. All are instance methods.
IN THIS SECTION:

- **addMessage(message)**
  Add a message to the current page context.

- **addMessages(exceptionThrown)**
  Adds a list of messages to the current page context based on a thrown exception.

- **currentPage()**
  Returns the current page's PageReference.

- **getMessages()**
  Returns a list of the messages associated with the current context.

- **hasMessages()**
  Returns `true` if there are messages associated with the current context, `false` otherwise.

- **hasMessages(severity)**
  Returns `true` if messages of the specified severity exist, `false` otherwise.

### addMessage (message)

Add a message to the current page context.

**Signature**

```java
public Void addMessage(ApexPages.Message message)
```

**Parameters**

- **message**
  Type: `ApexPages.Message`

**Return Value**

Type: `Void`

### addMessages (exceptionThrown)

Adds a list of messages to the current page context based on a thrown exception.

**Signature**

```java
public Void addMessages(Exception exceptionThrown)
```

**Parameters**

- **exceptionThrown**
  Type: `Exception`

**Return Value**

Type: `Void`
currentPage()
Returns the current page's PageReference.

Signature
public System.PageReference currentPage()

Return Value
Type: System.PageReference

Example
This code segment returns the id parameter of the current page.

```java
public MyController() {
    account = [
        SELECT Id, Name, Site
        FROM Account
        WHERE Id = :ApexPages.currentPage().
            getParameters().
            get('id')
    ];
}
```

getMessages()
Returns a list of the messages associated with the current context.

Signature
public ApexPages.Message[] getMessages()

Return Value
Type: ApexPages.Message[]

hasMessages()
Returns true if there are messages associated with the current context, false otherwise.

Signature
public Boolean hasMessages()

Return Value
Type: Boolean
hasMessages(severity)

Returns true if messages of the specified severity exist, false otherwise.

Signature

public Boolean hasMessages(ApexPages.Severity severity)

Parameters

severity

Type: ApexPages.Severity

Return Value

Type: Boolean

Approval Class

Contains methods for processing approval requests and setting approval-process locks and unlocks on records.

Namespace

System

Usage

Salesforce admins can edit locked records. Depending on your approval process configuration settings, an assigned approver can also edit locked records. Locks and unlocks that are set programmatically use the same record editability settings as other approval-process locks and unlocks.

Record locks and unlocks are treated as DML. They’re blocked before a callout, they count toward your DML limits, and if a failure occurs, they’re rolled back along with the rest of your transaction. To change this rollback behavior, use an allOrNone parameter.

Approval is also used as a namespace for the ProcessRequest and ProcessResult classes.

SEE ALSO:

Approval Process Considerations

Approval Methods

The following are methods for Approval. All methods are static.

IN THIS SECTION:

isLocked(id)

Returns true if the record with the ID id is locked, or false if it’s not.

isLocked(ids)

Returns a map of record IDs and their lock statuses. If the record is locked the status is true. If the record is not locked the status is false.
isLocked(sobject)
Returns true if the sobject record is locked, or false if it's not.
isLocked(sobjects)
Returns a map of record IDs to lock statuses. If the record is locked the status is true. If the record is not locked the status is false.
lock(recordId)
Locks an object, and returns the lock results.
lock(recordIds)
Locks a set of objects, and returns the lock results, including failures.
lock(recordToLock)
Locks an object, and returns the lock results.
lock(recordsToLock)
Locks a set of objects, and returns the lock results, including failures.
lock(recordId, allOrNothing)
Locks an object, with the option for partial success, and returns the lock result.
lock(recordIds, allOrNothing)
Locks a set of objects, with the option for partial success. It returns the lock results, including failures.
lock(recordToLock, allOrNothing)
Locks an object, with the option for partial success, and returns the lock result.
lock(recordsToLock, allOrNothing)
Locks a set of objects, with the option for partial success. It returns the lock results, including failures.
process(approvalRequest)
Submits a new approval request and approves or rejects existing approval requests.
process(approvalRequest, allOrNone)
Submits a new approval request and approves or rejects existing approval requests.
process(approvalRequests)
Submits a list of new approval requests, and approves or rejects existing approval requests.
process(approvalRequests, allOrNone)
Submits a list of new approval requests, and approves or rejects existing approval requests.
unlock(recordId)
Unlocks an object, and returns the unlock results.
unlock(recordIds)
Unlocks a set of objects, and returns the unlock results, including failures.
unlock(recordToUnlock)
Unlocks an object, and returns the unlock results.
unlock(recordsToUnlock)
Unlocks a set of objects, and returns the unlock results, including failures.
unlock(recordId, allOrNothing)
Unlocks an object, with the option for partial success, and returns the unlock result.
unlock(recordIds, allOrNothing)
Unlocks a set of objects, with the option for partial success. It returns the unlock results, including failures.
unlock(recordToUnlock, allOrNothing)
Unlocks an object, with the option for partial success, and returns the unlock result.

unlock(recordsToUnlock, allOrNothing)
Unlocks a set of objects, with the option for partial success. It returns the unlock results, including failures.

isLocked(id)
Returns true if the record with the ID id is locked, or false if it's not.

Signature
public static Boolean isLocked(Id id)

Parameters
id
Type: Id
The ID of the record whose lock or unlock status is in question.

Return Value
Type: Boolean

isLocked(ids)
Returns a map of record IDs and their lock statuses. If the record is locked the status is true. If the record is not locked the status is false.

Signature
public static Map<Id,Boolean> isLocked(List<Id> ids)

Parameters
ids
Type: List<Id>
The IDs of the records whose lock or unlock statuses are in question.

Return Value
Type: Map<Id,Boolean>

isLocked(sobject)
Returns true if the sobject record is locked, or false if it's not.

Signature
public static Boolean isLocked(SObject sobject)
Parameters

\texttt{sobject}

Type: SObject

The record whose lock or unlock status is in question.

Return Value

Type: Boolean

\texttt{isLocked(sobjects)}

Returns a map of record IDs to lock statuses. If the record is locked the status is \texttt{true}. If the record is not locked the status is \texttt{false}.

Signature

\texttt{public static Map<Id, Boolean> isLocked(List<SObject> sobjects)}

Parameters

\texttt{sobjects}

Type: List<SObject>

The records whose lock or unlock statuses are in question.

Return Value

Type: Map<Id, Boolean>

\texttt{lock(recordId)}

Locks an object, and returns the lock results.

Signature

\texttt{public static Approval.LockResult lock(Id recordId)}

Parameters

\texttt{recordId}

Type: Id

ID of the object to lock.

Return Value

Type: Approval.LockResult

\texttt{lock(recordIds)}

Locks a set of objects, and returns the lock results, including failures.
Signature

```java
public static List<Approval.LockResult> lock(List<Id> ids)
```

**Parameters**

`ids`
Type: `List<Id>`
IDs of the objects to lock.

**Return Value**
Type: `List<Approval.LockResult>`

`lock(recordToLock)`
Locks an object, and returns the lock results.

**Signature**

```java
public static Approval.LockResult lock(SObject recordToLock)
```

**Parameters**

`recordToLock`
Type: `SObject`

**Return Value**
Type: `Approval.LockResult`

`lock(recordsToLock)`
Locks a set of objects, and returns the lock results, including failures.

**Signature**

```java
public static List<Approval.LockResult> lock(List<SObject> recordsToLock)
```

**Parameters**

`recordsToLock`
Type: `List<SObject>`

**Return Value**
Type: `List<Approval.LockResult>`

`lock(recordId, allOrNothing)`
Locks an object, with the option for partial success, and returns the lock result.
Signature

public static Approval.LockResult lock(Id recordId, Boolean allOrNothing)

Parameters

recordId
Type: Id
ID of the object to lock.

allOrNothing
Type: Boolean
Specifies whether this operation allows partial success. If you specify false and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that you can use to verify which records succeeded, which failed, and why.

Return Value
Type: Approval.LockResult

lock(recordIds, allOrNothing)
Locks a set of objects, with the option for partial success. It returns the lock results, including failures.

Signature

public static List<Approval.LockResult> lock(List<Id> recordIds, Boolean allOrNothing)

Parameters

recordIds
Type: List<Id>
IDs of the objects to lock.

allOrNothing
Type: Boolean
Specifies whether this operation allows partial success. If you specify false and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that you can use to verify which records succeeded, which failed, and why.

Return Value
Type: List<Approval.LockResult>

lock(recordToLock, allOrNothing)
_locks an object, with the option for partial success, and returns the lock result.

Signature

public static Approval.LockResult lock(SObject recordToLock, Boolean allOrNothing)
Parameters

recordToLock
Type: SObject

allOrNothing
Type: Boolean

Specifies whether this operation allows partial success. If you specify false and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that you can use to verify which records succeeded, which failed, and why.

Return Value
Type: Approval.LockResult

lock(recordsToLock, allOrNothing)
Locks a set of objects, with the option for partial success. It returns the lock results, including failures.

Signature
public static List<Approval.LockResult> lock(List<SObject> recordsToLock, Boolean allOrNothing)

Parameters

recordsToLock
Type: List<SObject>

allOrNothing
Type: Boolean

Specifies whether this operation allows partial success. If you specify false and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that you can use to verify which records succeeded, which failed, and why.

Return Value
Type: List<Approval.LockResult>

process(approvalRequest)
Submits a new approval request and approves or rejects existing approval requests.

Signature
public static Approval.ProcessResult process(Approval.ProcessRequest approvalRequest)

Parameters

approvalRequest
Type: Approval.ProcessRequest
Return Value
Type: Approval.ProcessResult

Example

```java
// Insert an account
Account a = new Account(Name='Test',
    annualRevenue=100.0);
insert a;

// Create an approval request for the account
Approval.ProcessSubmitRequest req1 =
    new Approval.ProcessSubmitRequest();
req1.setObjectId(a.id);

// Submit the approval request for the account
Approval.ProcessResult result =
    Approval.process(req1);
```

`process(approvalRequest, allOrNone)`
Submits a new approval request and approves or rejects existing approval requests.

Signature

```java
public static Approval.ProcessResult process(Approval.ProcessRequest approvalRequest,
    Boolean allOrNone)
```

Parameters

- `approvalRequest` Approval.ProcessRequest
- `allOrNone` Type: Boolean
  The optional `allOrNone` parameter specifies whether the operation allows for partial success. If you specify `false` for this parameter and an approval fails, the remainder of the approval processes can still succeed.

Return Value
Approval.ProcessResult

`process(approvalRequests)`
Submits a list of new approval requests, and approves or rejects existing approval requests.
### process(approvalRequests, allOrNone)

Submits a list of new approval requests, and approves or rejects existing approval requests.

#### Signature

```java
public static Approval.ProcessResult[] process(Approval.ProcessRequest[] approvalRequests, Boolean allOrNone)
```

#### Parameters

- `approvalRequests`
  - Type: `Approval.ProcessRequest[]`

- `allOrNone`
  - Type: `Boolean`
  - The optional `allOrNone` parameter specifies whether the operation allows for partial success. If you specify `false` for this parameter and an approval fails, the remainder of the approval processes can still succeed.

#### Return Value

- `Approval.ProcessResult[]`

### unlock(recordId)

Unlocks an object, and returns the unlock results.

#### Signature

```java
public static Approval.UnlockResult unlock(Id recordId)
```

#### Parameters

- `recordId`
  - Type: `Id`
  - ID of the object to unlock.
Return Value
Type: Approval.UnlockResult

unlock(recordIds)
Unlocks a set of objects, and returns the unlock results, including failures.

Signature
public static List<Approval.UnlockResult> unlock(List<Id> recordIds)

Parameters
recordIds
Type: List<Id>
IDs of the objects to unlock.

Return Value
Type: List<Approval.UnlockResult>

unlock(recordToUnlock)
Unlocks an object, and returns the unlock results.

Signature
public static Approval.UnlockResult unlock(SObject recordToUnlock)

Parameters
recordToUnlock
Type: SObject

Return Value
Type: Approval.UnlockResult

unlock(recordsToUnlock)
Unlocks a set of objects, and returns the unlock results, including failures.

Signature
public static List<Approval.UnlockResult> unlock(List<SObject> recordsToUnlock)

Parameters
recordsToUnlock
Type: List<SObject>
Return Value
Type: List<Approval.UnlockResult>

unlock(recordId, allOrNothing)
Unlocks an object, with the option for partial success, and returns the unlock result.

Signature
public static Approval.UnlockResult unlock(Id recordId, Boolean allOrNothing)

Parameters
recordId
Type: Id
ID of the object to lock.

allOrNothing
Type: Boolean
Specifies whether this operation allows partial success. If you specify false and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that you can use to verify which records succeeded, which failed, and why.

Return Value
Type: Approval.UnlockResult

unlock(recordIds, allOrNothing)
Unlocks a set of objects, with the option for partial success. It returns the unlock results, including failures.

Signature
public static List<Approval.UnlockResult> unlock(List<Id> recordIds, Boolean allOrNothing)

Parameters
recordIds
Type: List<Id>
IDs of the objects to unlock.

allOrNothing
Type: Boolean
Specifies whether this operation allows partial success. If you specify false and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that you can use to verify which records succeeded, which failed, and why.

Return Value
Type: List<Approval.UnlockResult>
unlock(recordToUnlock, allOrNothing)
Unlocks an object, with the option for partial success, and returns the unlock result.

Signature
public static Approval.UnlockResult unlock(SObject recordToUnlock, Boolean allOrNothing)

Parameters
recordToUnlock
  Type: SObject
allOrNothing
  Type: Boolean
  Specifies whether this operation allows partial success. If you specify false and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that you can use to verify which records succeeded, which failed, and why.

Return Value
Type: Approval.UnlockResult

unlock(recordsToUnlock, allOrNothing)
Unlocks a set of objects, with the option for partial success. It returns the unlock results, including failures.

Signature
public static List<Approval.UnlockResult> unlock(List<SObject> recordsToUnlock, Boolean allOrNothing)

Parameters
recordsToUnlock
  Type: List<SObject>
allOrNothing
  Type: Boolean
  Specifies whether this operation allows partial success. If you specify false and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that you can use to verify which records succeeded, which failed, and why.

Return Value
Type: List<Approval.UnlockResult>

Assert Class
Contains methods to assert various conditions with test methods, such as whether two values are the same, a condition is true, or a variable is null.
Namespace

System

Assert Methods

The following are methods for Assert.

IN THIS SECTION:

- `areEqual(expected, actual, msg)`
  Asserts that the first two arguments are the same.
- `areEqual(expected, actual)`
  Asserts that the two arguments are the same.
- `areNotEqual(notExpected, actual, msg)`
  Asserts that the first two arguments aren’t the same.
- `areNotEqual(notExpected, actual)`
  Asserts that the two arguments aren’t the same.
- `fail(msg)`
  Immediately return a fatal error that causes code execution to halt.
- `fail()`
  Immediately return a fatal error that causes code execution to halt.
- `isFalse(condition, msg)`
  Asserts that the specified condition is false.
- `isFalse(condition)`
  Asserts that the specified condition is false.
- `isInstanceOfType(instance, expectedType, msg)`
  Asserts that the instance is of the specified type.
- `isInstanceOfType(instance, expectedType)`
  Asserts that the instance is of the specified type.
- `isNotInstanceOfType(instance, notExpectedType, msg)`
  Asserts that the instance isn’t of the specified type.
- `isNotInstanceOfType(instance, notExpectedType)`
  Asserts that the instance isn’t of the specified type.
- `isNotNull(value, msg)`
  Asserts that the value isn’t null.
- `isNotNull(value)`
  Asserts that the value isn’t null.
- `isNull(value, msg)`
  Asserts that the value is null.
- `isNull(value)`
  Asserts that the value is null.
**isTrue(condition, msg)**
Asserts that the specified condition is true.

**isTrue(condition)**
Asserts that the specified condition is true.

**areEqual(expected, actual, msg)**
Asserts that the first two arguments are the same.

**Signature**
```
public static void areEqual(Object expected, Object actual, String msg)
```

**Parameters**
- **expected**
  - Type: Object
  - Expected value.
- **actual**
  - Type: Object
  - Actual value.
- **msg**
  - Type: String
  - (Optional) Custom message returned as part of the error message.

**Return Value**
Type: void

**Usage**
If the first two arguments aren't the same, a fatal error is returned that causes code execution to halt.
You can't catch an assertion failure using a try/catch block even though it's logged as an exception.

**Example**
```
String sub = 'abcde'.substring(2);
Assert.areEqual('cde', sub, 'Expected characters after first two'); // Succeeds
```

**areEqual(expected, actual)**
Asserts that the two arguments are the same.

**Signature**
```
public static void areEqual(Object expected, Object actual)
```
Parameters

expected
  Type: Object
  Expected value.

actual
  Type: Object
  Actual value.

Return Value
Type: void

Usage
If the two arguments aren’t the same, a fatal error is returned that causes code execution to halt.

You can’t catch an assertion failure using a try/catch block even though it’s logged as an exception.

Example

```java
String sub = 'abcde'.substring(2);
Assert.areEqual('cde', sub); // Succeeds
```

areNotEqual(notExpected, actual, msg)

Asserts that the first two arguments aren’t the same.

Signature

```java
public static void areNotEqual(Object notExpected, Object actual, String msg)
```

Parameters

notExpected
  Type: Object
  Value that’s not expected.

actual
  Type: Object
  Actual value.

msg
  Type: String
  (Optional) Custom message returned as part of the error message.

Return Value
Type: void
Usage
If the first two arguments are the same, a fatal error is returned that causes code execution to halt.
You can’t catch an assertion failure using a try/catch block even though it’s logged as an exception.

Example

```java
String sub = 'abcde'.substring(2);
Assert.areNotEqual('xyz', sub, 'Characters not expected after first two'); // Succeeds
```

**areNotEqual(notExpected, actual)**

Asserts that the two arguments aren’t the same.

**Signature**

```java
public static void areNotEqual(Object notExpected, Object actual)
```

**Parameters**

- **notExpected**
  - Type: Object
  - Value that’s not expected.

- **actual**
  - Type: Object
  - Actual value.

**Return Value**

Type: void

Usage

If the two arguments are the same, a fatal error is returned that causes code execution to halt.
You can’t catch an assertion failure using a try/catch block even though it’s logged as an exception.

Example

```java
String sub = 'abcde'.substring(2);
Assert.areNotEqual('xyz', sub); // Succeeds
```

**fail(msg)**

Immediately return a fatal error that causes code execution to halt.

**Signature**

```java
public static void fail(String msg)
```
Parameters

msg
Type: String
(Optional) Custom message returned as part of the error message.

Return Value
Type: void

Usage
Commonly used in a try/catch block test case where an exception is expected to be thrown. You can't, however, catch the assertion failure in the try/catch block even though it's logged as an exception.

Example

```java
// test case where exception is expected
try {
    SomeClass.methodUnderTest();
    Assert.fail('DmlException Expected');
} catch (DmlException ex) {
    // Add assertions here about the expected exception
}
```

fail()
Immediately return a fatal error that causes code execution to halt.

Signature

public static void fail()

Return Value
Type: void

Usage
Commonly used in a try/catch block test case where an exception is expected to be thrown. You can't, however, catch the assertion failure in the try/catch block even though it's logged as an exception.

Example

```java
// test case where exception is expected
try {
    SomeClass.methodUnderTest();
    Assert.fail();
} catch (DmlException ex) {
    // Add assertions here about the expected exception
}
```
isFalse(condition, msg)

Asserts that the specified condition is false.

Signature

public static void isFalse(Boolean condition, String msg)

Parameters

condition
Type: Boolean
Condition you're checking to determine if it's false.

msg
Type: String
(Optional) Custom message returned as part of the error message.

Return Value

Type: void

Usage

If the condition is true, a fatal error is returned that causes code execution to halt.
You can't catch an assertion failure using a try/catch block even though it's logged as an exception.

Example

Boolean containsCode = 'Salesforce'.contains('code');
Assert.isFalse(containsCode, 'No code'); // Assertion succeeds

isFalse(condition)

Asserts that the specified condition is false.

Signature

public static void isFalse(Boolean condition)

Parameters

condition
Type: Boolean
Condition you're checking to determine if it's false.

Return Value

Type: void
Usage
If the condition is `true`, a fatal error is returned that causes code execution to halt.
You can’t catch an assertion failure using a try/catch block even though it’s logged as an exception.

Example
```java
Boolean containsCode = 'Salesforce'.contains('code');
Assert.isFalse(containsCode); // Assertion succeeds
```

`isInstanceOfType(instance, expectedType, msg)`
Asserts that the instance is of the specified type.

Signature
```java
public static void isInstanceOfType(Object instance, System.Type expectedType, String msg)
```

Parameters
- `instance`
  - Type: Object
  - Instance whose type you’re checking.
- `expectedType`
  - Type: `System.Type` on page 3532
  - Expected type.
- `msg`
  - Type: String
  - (Optional) Custom message returned as part of the error message.

Return Value
Type: void

Usage
If the instance isn’t of the specified type, a fatal error is returned that causes code execution to halt.
You can’t catch an assertion failure using a try/catch block even though it’s logged as an exception.

Example
```java
Account o = new Account();
Assert.isInstanceOfType(o, Account.class); // Succeeds
```

`isInstanceOfType(instance, expectedType)`
Asserts that the instance is of the specified type.
### isInstanceOfType

#### Signature

```
public static void isInstanceOfType(Object instance, System.Type expectedType)
```

#### Parameters

- `instance`  
  Type: Object  
  Instance whose type you’re checking.

- `expectedType`  
  Type: System.Type on page 3532  
  Expected type.

#### Return Value

Type: void

#### Usage

If the instance isn’t of the specified type, a fatal error is returned that causes code execution to halt.

You can’t catch an assertion failure using a try/catch block even though it’s logged as an exception.

#### Example

```
Account o = new Account();
Assert.isInstanceOfType(o, Account.class); // Succeeds
```

```
Account o = new Account();
Assert.isInstanceOfType(o, Account.class, 'Expected type.'); // Succeeds
```

### isNotInstanceOfType

#### Signature

```
public static void isNotInstanceOfType(Object instance, System.Type notExpectedType, String msg)
```

#### Parameters

- `instance`  
  Type: Object  
  Instance whose type you’re checking.

- `notExpectedType`  
  Type: System.Type on page 3532  
  Type that’s not expected.

#### Description

`isNotInstanceOfType` asserts that the instance isn’t of the specified type.
msg
  Type: String
  (Optional) Custom message returned as part of the error message.

Return Value
Type: void

Usage
If the instance is of the specified type, a fatal error is returned that causes code execution to halt.
You can’t catch an assertion failure using a try/catch block even though it’s logged as an exception.

Example
Contact con = new Contact();
Assert.isNotInstanceOfType(con, Account.class, 'Not expected type'); // Succeeds

isNotInstanceOfType(instance, notExpectedType)
Asserts that the instance isn’t of the specified type.

Signature
public static void isNotInstanceOfType(Object instance, System.Type notExpectedType)

Parameters
instance
  Type: Object
  Instance whose type you’re checking.
notExpectedType
  Type: System.Type
  Type that’s not expected.

Return Value
Type: void

Usage
If the instance is of the specified type, a fatal error is returned that causes code execution to halt.
You can’t catch an assertion failure using a try/catch block even though it’s logged as an exception.

Example
Contact con = new Contact();
Assert.isNotInstanceOfType(con, Account.class); // Succeeds
**isNotNull(value, msg)**

Asserts that the value isn’t null.

**Signature**

```java
public static void isNotNull(Object value, String msg)
```

**Parameters**

- **value**
  - Type: `Object`
  - Value you’re checking to determine if it’s not null.

- **msg**
  - Type: `String`
  - (Optional) Custom message returned as part of the error message.

**Return Value**

Type: `void`

**Usage**

If the value is null, a fatal error is returned that causes code execution to halt.

You can’t catch an assertion failure using a try/catch block even though it’s logged as an exception.

**Example**

```java
String myString = 'value';
Assert.isNotNull(myString, 'myString should not be null'); // Succeeds
```

**isNotNull(value)**

Asserts that the value isn’t null.

**Signature**

```java
public static void isNotNull(Object value)
```

**Parameters**

- **value**
  - Type: `Object`
  - Value you’re checking to determine if it’s not null.

**Return Value**

Type: `void`
Usage
If the value is null, a fatal error is returned that causes code execution to halt.
You can’t catch an assertion failure using a try/catch block even though it’s logged as an exception.

Example
String myString = 'value';
Assert.isNotNull(myString); // Succeeds

isNull(value, msg)
Asserts that the value is null.

Signature
public static void isNull(Object value, String msg)

Parameters
value
Type: Object
Value you’re checking to determine if it’s null.

msg
Type: String
(Optional) Custom message returned as part of the error message.

Return Value
Type: void

Usage
If the value isn’t null, a fatal error is returned that causes code execution to halt.
You can’t catch an assertion failure using a try/catch block even though it’s logged as an exception.

Example
String myString = null;
Assert.isNull(myString, 'String should be null'); // Succeeds

isNull(value)
Asserts that the value is null.

Signature
public static void isNull(Object value)
Parameters
value
Type: Object
Value you’re checking to determine if it’s null.

Return Value
Type: void

Usage
If the value isn’t null, a fatal error is returned that causes code execution to halt.
You can’t catch an assertion failure using a try/catch block even though it’s logged as an exception.

Example
String myString = null;
Assert.isNull(myString); // Succeeds

isTrue(condition, msg)
Asserts that the specified condition is true.

Signature
public static void isTrue(Boolean condition, String msg)

Parameters
condition
Type: Boolean
Condition you’re checking to determine if it’s true.

msg
Type: String
(Optional) Custom message returned as part of the error message.

Return Value
Type: void

Usage
If the specified condition is false, a fatal error is returned that causes code execution to halt.
You can’t catch an assertion failure using a try/catch block even though it’s logged as an exception.
Example

```java
Boolean containsForce = 'Salesforce'.contains('force');
Assert.isTrue(containsForce, 'Contains force'); // Assertion succeeds
```

**isTrue(condition)**

Asserts that the specified condition is true.

**Signature**

```java
public static void isTrue(Boolean condition)
```

**Parameters**

- `condition`  
  Type: `Boolean`  
  Condition you're checking to determine if it's true.

**Return Value**

Type: `void`

**Usage**

If the specified condition is `false`, a fatal error is returned that causes code execution to halt.

You can't catch an assertion failure using a try/catch block even though it's logged as an exception.

Example

```java
Boolean containsForce = 'Salesforce'.contains('force');
Assert.isTrue(containsForce); // Assertion succeeds
```

**AsyncInfo Class**

Provides methods to get the current stack depth, maximum stack depth, and the minimum queueable delay for Queueable transactions, and to determine if maximum stack depth is set.

**Namespace**

`System`

**AsyncInfo Methods**

The following are methods for `AsyncInfo`. 
IN THIS SECTION:

- `getCurrentQueueableStackDepth()`
  Get the current queueable stack depth for queueable transactions.

- `getMaximumQueueableStackDepth()`
  Get the maximum queueable stack depth for queueable transactions.

- `getMinimumQueueableDelayInMinutes()`
  Get the minimum queueable delay for queueable transactions (in minutes).

- `hasMaxStackDepth()`
  Determine if maximum stack depth is set for your queueable requests.

---

**getCurrentQueueableStackDepth()**

Get the current queueable stack depth for queueable transactions.

**Signature**

```java
public static Integer getCurrentQueueableStackDepth()
```

**Return Value**

Type: `Integer`

---

**getMaximumQueueableStackDepth()**

Get the maximum queueable stack depth for queueable transactions.

**Signature**

```java
public static Integer getMaximumQueueableStackDepth()
```

**Return Value**

Type: `Integer`

---

**getMinimumQueueableDelayInMinutes()**

Get the minimum queueable delay for queueable transactions (in minutes).

**Signature**

```java
public static Integer getMinimumQueueableDelayInMinutes()
```

**Return Value**

Type: `Integer`

Returns null if no delay is defined.
**hasMaxStackDepth()**
Determine if maximum stack depth is set for your queueable requests.

**Signature**
public static Boolean hasMaxStackDepth()

**Return Value**
Type: Boolean

**AsyncOptions Class**
Contains maximum stack depths for queueable transactions and the minimum queueable delay in minutes. Passed as parameter to the `System.enqueueJob()` method to define a unique queueable job signature, the maximum stack depth for queueable transactions and the minimum queueable delay in minutes.

**Namespace**
System

**AsyncOptions Properties**
The following are properties for AsyncOptions.

**DuplicateSignature**
A unique signature for a Queueable job.

**MaximumQueueableStackDepth**
Maximum stack depth for queueable transactions.

**MinimumQueueableDelayInMinutes**
Minimum queueable delay for queueable transactions.

**DuplicateSignature**
A unique signature for a Queueable job.

**Signature**
public System.QueueableDuplicateSignature DuplicateSignature {get; set;}
Property Value
Type: QueueableDuplicateSignature Class

**MaximumQueueableStackDepth**
Maximum stack depth for queueable transactions.

Signature
public Integer MaximumQueueableStackDepth {get; set;}

Property Value
Type: Integer

**MinimumQueueableDelayInMinutes**
Minimum queueable delay for queueable transactions.

Signature
public Integer MinimumQueueableDelayInMinutes {get; set;}

Property Value
Type: Integer

**Blob Class**
Contains methods for the Blob primitive data type.

**Namespace**
System

**Usage**
For more information on Blobs, see Primitive Data Types.

**Blob Methods**
The following are methods for Blob.

IN THIS SECTION:
- size()
  - Returns the number of characters in the Blob.
- toPdf(stringToConvert)
  - Creates a binary object out of the given string, encoding it as a PDF file.
toString()
Casts the Blob into a String.

valueOf(stringToBlob)
Casts the specified String to a Blob.

size()
Returns the number of characters in the Blob.

Signature
public Integer size()

Return Value
Type: Integer

Example
```java
String myString = 'StringToBlob';
Blob myBlob = Blob.valueOf(myString);
Integer size = myBlob.size();
```

toPdf(stringToConvert)
Creates a binary object out of the given string, encoding it as a PDF file.

Signature
public static Blob toPdf(String stringToConvert)

Parameters
stringToConvert
Type: String

Note: Referencing a static resource throws an InvalidParameterValue exception.

Return Value
Type: Blob

Example
```java
String pdfContent = 'This is a test string';
Account a = new account(name = 'test');
insert a;
Attachment attachmentPDF = new Attachment();
attachmentPdf.parentId = a.id;
attachmentPdf.name = a.name + '.pdf';
```
**attachmentPdf.body** = blob.toPDF(pdfContent);
insert attachmentPDF;

**toString()**
Casts the Blob into a String.

**Signature**
public String toString()

**Return Value**
Type: String

**Example**
```java
String myString = 'StringToBlob';
Blob myBlob = Blob.valueOf(myString);
System.assertEquals('StringToBlob', myBlob.toString());
```

**valueOf (stringToBlob)**
Casts the specified String to a Blob.

**Signature**
public static Blob valueOf(String stringToBlob)

**Parameters**
stringToBlob
Type: String

**Return Value**
Type: Blob

**Example**
```java
String myString = 'StringToBlob';
Blob myBlob = Blob.valueOf(myString);
```

**Boolean Class**
Contains methods for the Boolean primitive data type.
Namespace
System

Boolean Methods
The following are methods for Boolean. All methods are static.

IN THIS SECTION:
 valueOf(stringToBoolean)
Converts the specified string to a Boolean value and returns true if the specified string value is true. Otherwise, returns false.
 valueOf(fieldValue)
Converts the specified object to a Boolean value. Use this method to convert a history tracking field value or an object that represents a Boolean value.

valueOf(stringToBoolean)
Converts the specified string to a Boolean value and returns true if the specified string value is true. Otherwise, returns false.

Signature
public static Boolean valueOf(String stringToBoolean)

Parameters
stringToBoolean
Type: String

Return Value
Type: Boolean

Usage
If the specified argument is null, this method throws an exception.

Example
```java
Boolean b = Boolean.valueOf('true');
System.assertEquals(true, b);
```

valueOf(fieldValue)
Converts the specified object to a Boolean value. Use this method to convert a history tracking field value or an object that represents a Boolean value.

Signature
public static Boolean valueOf(Object fieldValue)
Parameters

`fieldValue`  
Type: Object

Return Value

Type: Boolean

Usage

Use this method with the `OldValue` or `NewValue` fields of history sObjects, such as `AccountHistory`, when the field type corresponds to a Boolean type, like a checkbox field.

Example

```java
List<AccountHistory> ahlist =
    [SELECT Field, OldValue, NewValue
     FROM AccountHistory];
for(AccountHistory ah : ahlist) {
    System.debug('Field: ' + ah.Field);
    if (ah.field == 'IsPlatinum__c') {
        Boolean oldValue =
            Boolean.valueOf(ah.OldValue);
        Boolean newValue =
            Boolean.valueOf(ah.NewValue);
    }
}
```

BusinessHours Class

Use the `BusinessHours` methods to set the business hours at which your customer support team operates.

Namespace

`System`

BusinessHours Methods

The following are methods for `BusinessHours`. All methods are static.

IN THIS SECTION:

- `add-businessHoursId, startDate, intervalMilliseconds)`
  Adds an interval of time from a start Datetime traversing business hours only. Returns the result Datetime in the local time zone.
- `addGmt-businessHoursId, startDate, intervalMilliseconds)`
  Adds an interval of milliseconds from a start Datetime traversing business hours only. Returns the result Datetime in GMT.
- `diff-businessHoursId, startDate, endDate)`
  Returns the difference in milliseconds between a start and end Datetime based on a specific set of business hours.
isWithin(businessHoursId, targetDate)
Returns true if the specified target date occurs within business hours. Holidays are included in the calculation.

nextStartDate(businessHoursId, targetDate)
Starting from the specified target date, returns the next date when business hours are open. If the specified target date falls within business hours, this target date is returned.

add(businessHoursId, startDate, intervalMilliseconds)
Adds an interval of time from a start Datetime traversing business hours only. Returns the result Datetime in the local time zone.

Signature
public static Datetime add(String businessHoursId, Datetime startDate, Long intervalMilliseconds)

Parameters
businessHoursId
Type: String
startDate
Type: Datetime
intervalMilliseconds
Type: Long
Interval value should be provided in milliseconds, however time precision smaller than one minute is ignored.

Return Value
Type: Datetime

addGmt(businessHoursId, startDate, intervalMilliseconds)
Adds an interval of milliseconds from a start Datetime traversing business hours only. Returns the result Datetime in GMT.

Signature
public static Datetime addGmt(String businessHoursId, Datetime startDate, Long intervalMilliseconds)

Parameters
businessHoursId
Type: String
startDate
Type: Datetime
intervalMilliseconds
Type: Long
**diff(businessHoursId, startDate, endDate)**

Returns the difference in milliseconds between a start and end Datetime based on a specific set of business hours.

**Signature**

```java
public static Long diff(String businessHoursId, Datetime startDate, Datetime endDate)
```

**Parameters**

- `businessHoursId`
  - Type: String
- `startDate`
  - Type: Datetime
- `endDate`
  - Type: Datetime

**Return Value**

Type: Long

**isWithin(businessHoursId, targetDate)**

Returns `true` if the specified target date occurs within business hours. Holidays are included in the calculation.

**Signature**

```java
public static Boolean isWithin(String businessHoursId, Datetime targetDate)
```

**Parameters**

- `businessHoursId`
  - Type: String
  - The business hours ID.
- `targetDate`
  - Type: Datetime
  - The date to verify.

**Return Value**

Type: Boolean
Example
The following example finds whether a given time is within the default business hours.

```java
// Get the default business hours
BusinessHours bh = [SELECT Id FROM BusinessHours WHERE IsDefault=true];

// Create Datetime on May 28, 2013 at 1:06:08 AM in the local timezone.
Datetime targetTime = Datetime.newInstance(2013, 5, 28, 1, 6, 8);

// Find whether the time is within the default business hours
Boolean isWithin = BusinessHours.isWithin(bh.id, targetTime);
```

**nextStartDate(businessHoursId, targetDate)**

Starting from the specified target date, returns the next date when business hours are open. If the specified target date falls within business hours, this target date is returned.

**Signature**

```java
public static Datetime nextStartDate(String businessHoursId, Datetime targetDate)
```

**Parameters**

- **businessHoursId**
  - Type: `String`
  - The business hours ID.

- **targetDate**
  - Type: `Datetime`
  - The date used as a start date to obtain the next date.

**Return Value**

Type: `Datetime`

**Example**

The following example finds the next date starting from the target date when business hours reopens. If the target date is within the given business hours, the target date is returned. The returned time is in the local time zone.

```java
// Get the default business hours
BusinessHours bh = [SELECT Id FROM BusinessHours WHERE IsDefault=true];

// Create Datetime on May 28, 2013 at 1:06:08 AM in the local timezone.
Datetime targetTime = Datetime.newInstance(2013, 5, 28, 1, 6, 8);
// Starting from the targetTime, find the next date when business hours reopens. Return the target time.

// if it is within the business hours. The returned time will be in the local time zone
Datetime nextStart = BusinessHours.nextStartDate(bh.id, targetTime);
```
Callable Interface

Enables developers to use a common interface to build loosely coupled integrations between Apex classes or triggers, even for code in separate packages. Agreeing upon a common interface enables developers from different companies or different departments to build upon one another’s solutions. Implement this interface to enable the broader community, which might have different solutions than the ones you had in mind, to extend your code’s functionality.

Note: This interface is not an analog of the Java Callable interface, which is used for asynchronous invocation. Don’t confuse the two.

Namespace
System

Usage
To implement the Callable interface, you need to write only one method: `call(String action, Map<String, Object> args)`.

In code that utilizes or tests an implementation of Callable, cast an instance of your type to Callable.

This interface is not intended to replace defining more specific interfaces. Rather, the Callable interface allows integrations in which code from different classes or packages can use common base types.

IN THIS SECTION:
Callable Methods
Callable Example Implementation

Callable Methods

The following are methods for Callable.

IN THIS SECTION:
call(action, args)
Provides functionality that other classes or packages can utilize and build upon.

call(action, args)
Provides functionality that other classes or packages can utilize and build upon.

Signature

public Object call(String action, Map<String, Object> args)

Parameters

action
Type: String
The behavior for the method to exhibit.
**args**

Type: Map on page 3198<String,Object>

Arguments to be used by the specified action.

**Return Value**

Type: Object

The result of the method invocation.

**Callable Example Implementation**

This class is an example implementation of the `System.Callable` interface.

```java
public class Extension implements Callable {

    // Actual method
    String concatStrings(String stringValue) {
        return stringValue + stringValue;
    }

    // Actual method
    Decimal multiplyNumbers(Decimal decimalValue) {
        return decimalValue * decimalValue;
    }

    // Dispatch actual methods
    public Object call(String action, Map<String, Object> args) {
        switch (action) {
            case 'concatStrings':
                return this.concatStrings((String)args.get('stringValue'));
            case 'multiplyNumbers':
                return this.multiplyNumbers((Decimal)args.get('decimalValue'));
            default:
                throw new ExtensionMalformedCallException('Method not implemented');
        }
    }

    public class ExtensionMalformedCallException extends Exception {}
}
```

The following test code illustrates how calling code utilizes the interface to call a method.

```java
@IsTest
private with sharing class ExtensionCaller {

    @IsTest
    private static void givenConfiguredExtensionWhenCalledThenValidResult() {

        // Given
        String extensionClass = 'Extension'; // Typically set via configuration
```

2861
Decimal decimalTestValue = 10;

// When
Callable extension =
    (Callable) Type.forName(extensionClass).newInstance();
Decimal result = (Decimal)
    extension.call('multiplyNumbers', new Map<String, Object> {
        'decimalValue' => decimalTestValue
    });

// Then
System.assertEquals(100, result);
}

SEE ALSO:

Apex Developer Guide: Classes and Casting

Cases Class

Use the Cases class to interact with case records.

Namespace

System

Cases Methods

The following are static methods for Cases.

IN THIS SECTION:

generateThreadingMessageId(caseId)
Returns an RFC 2822-compliant message identifier that contains information used to match the email and its replies to a case.

getCaseIdFromEmailHeaders(headers)
Returns the case ID corresponding to the specified email header information, or returns null if none is found.

getCaseIdFromEmailThreadId(emailThreadId)
Returns the case ID corresponding to the specified email thread ID. (Deprecated. Use getCaseIdFromEmailHeaders and EmailMessages.getRecordIdFromEmail instead.)

generateThreadingMessageId(caseId)
Returns an RFC 2822-compliant message identifier that contains information used to match the email and its replies to a case.

Signature

public static String generateThreadingMessageId(Id caseId)
Parameters

caseId
   Type: Id

   The case SObject ID to which replies to this email should be attached.

Return Value

Type: String

Usage

Use the returned message identifier when sending case-related emails in Apex. The returned message identifier can be used in Message-ID or References headers. However, because Salesforce doesn’t let users specify the Message-ID, we set this identifier in the References header. When users reply to the sent email, replies should be attached to the specified case.

Example

In this sample, we create an email with a message identifier so that the email and any responses can be associated with the related case.

```java
//Get your Case ID. Here we use a dummy ID
ID caseId = Id.valueOf('500xx000000bpkTAAQ');
//Create a SingleEmailMessage object
Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();
//Set recipients and other fields
email.setToAddresses(new String[] {'test@salesforce.com'});
email.setPlainTextBody('Test Email Notification');
//........... more fields ...........
//Get the threading message identifier
String messageId = Cases.generateThreadingMessageId(caseId);
//Insert the message identifier into the References header
email.setReferences(messageId);
//Send out the email
Messaging.sendEmail(new Messaging.SingleEmailMessage[]{email});
```

getCaseIdFromEmailHeaders(headers)

Returns the case ID corresponding to the specified email header information, or returns null if none is found.

Signature

```java
public static Id getCaseIdFromEmailHeaders(List<Messaging.InboundEmail.Header> headers)
```

Parameters

headers
   Type: List<Messaging.InboundEmail.Header>

Return Value

Type: Id
Usage

To optimize finding a match between email threads and cases in your custom code, we recommend that you use this method and EmailMessages.getRecordIdFromEmail to implement a combination of token- and header-based threading.

If you are transitioning from Ref ID threading, we recommend that you replace Cases.getCaseIdFromEmailThreadId with a combination of Cases.getCaseIdFromEmailHeaders and EmailMessages.getRecordIdFromEmail. If you choose to implement header-based threading only, replace Cases.getCaseIdFromEmailThreadId with Cases.getCaseIdFromEmailHeaders.

The headers argument is used to find the matching Case Id using values for the In-Reply-To and References headers based on RFC 2822. If Email-to-Case can’t find any emails with a matching In-Reply-To or References header, it also checks the incoming email for an Outlook-specific header called Thread-Index. The first 22 bytes of this header uniquely identify the thread. If Email-to-Case detects a Thread-Index header on the incoming mail, it looks for matching information in the ClientThreadIdentifier field in EmailMessage records. If a match is found, the customer’s reply email is linked to the related case.

Typically this method is used in Email Services so that you can provide your own handling of inbound emails using Apex code.

Example

If you implement header-based threading in your Email Services currently, we recommend that you use Lightning threading, which combines token-based threading and header-based threading. For header-based threading to continue to work, store emails as EmailMessage records with the MessagedIdentifier field set properly. With Lightning threading, you can use threading tokens as the primary threading method and rely on header-based threading as a fallback, or vice versa.

In this example, we rely on threading tokens and use header-based threading as a fallback.

```apex
global class AttachEmailMessageToCaseExample implements Messaging.InboundEmailHandler {
    global Messaging.InboundEmailResult handleInboundEmail(Messaging.inboundEmail email,
        Messaging.InboundEnvelope env) {
        // Create an InboundEmailResult object for returning the result of the
        // Apex Email Service.
        Messaging.InboundEmailResult result = new Messaging.InboundEmailResult();

        // Try to find the Case ID using threading tokens in email attributes.
        Id caseId = EmailMessages.getRecordIdFromEmail(email.subject, email.plainTextBody,
            email.htmlBody);

        // If we haven’t found the Case ID, try finding it using headers.
        if (caseId == null) {
            caseId = Cases.getCaseIdFromEmailHeaders(email.headers);
        }

        // If a Case isn’t found, create a new Case record.
        if (caseId == null) {
            Case c = new Case(Subject = email.subject);
            insert c;
            System.debug('New Case Object: ' + c);
            caseId = c.Id;
        }

        // Process recipients
        String toAddresses;
        if (email.toAddresses != null) {
```
toAddresses = String.join(email.toAddresses, '; ');

EmailMessage em = new EmailMessage(
    Status = '0',
    MessageIdentifier = email.messageId,
    ParentId = caseId,
    // Other important fields.
    FromAddress = email.fromAddress,
    FromName = email.fromName,
    ToAddress = toAddresses,
    TextBody = email.plainTextBody,
    HtmlBody = email.htmlBody,
    Subject = email.subject,
    // Parse thread-index header to remain consistent with Email-to-Case.
    ClientThreadIdentifier = getClientThreadIdentifier(email.headers)
    // Other fields you wish to add.
);

// Insert the new EmailMessage.
insert em;
System.debug('New EmailMessage Object: ' + em);

// Set the result to true. No need to send an email back to the user
// with an error message.
result.success = true;

// Return the result for the Apex Email Service.
return result;

private String getClientThreadIdentifier(List<Messaging.InboundEmail.Header> headers) {
    if (headers == null || headers.size() == 0) return null;
    try {
        for (Messaging.InboundEmail.Header header : headers) {
            if (header.name.equalsIgnoreCase('thread-index')) {
                Blob threadIndex = EncodingUtil.base64Decode(header.value.trim());
                return EncodingUtil.convertToHex(threadIndex).substring(0, 44).toUpperCase();
            }
        }
    } catch (Exception e) {
        return null;
    }
    return null;
}
**getCaseIdFromEmailThreadId(emailThreadId)**

Returns the case ID corresponding to the specified email thread ID. *(Deprecated. Use getCaseIdFromEmailHeaders and EmailMessages.getRecordIdFromEmail instead.)*

**Signature**

``` java
public static ID getCaseIdFromEmailThreadId(String emailThreadId)
```

**Parameters**

- `emailThreadId`
  
  Type: *String*

**Return Value**

Type: *ID*

**Usage**

The argument for `emailThreadId`, also known as Ref ID, has the format `!00Dxx01gEW.!500xx0Yktl`. This format was introduced in the Winter ’24 release. The previous format, `_00Dxx1gEW._500xxYktl`, is supported for backward compatibility, but emails sent from the Winter ’24 release onward use the new format. Other formats that include ref: or [ref: aren’t supported by this method.

**Collator Class**

Contains methods to get locale-specific instances that can be used for comparisons and sorting. Use the `getInstance()` method to obtain the Collator instance for a given locale and pass the Collator as the Comparator parameter to the `list.sort()` method.

**Namespace**

*System*

**Usage**

Because locale-sensitive sorting can produce different results depending on the user running the code, avoid using it in triggers or in code that expects a particular sort order.

**Example**

This example performs a default list sort and then uses Collator to sort based on the user locale.

``` java
@IsTest
static void userLocaleSort() {
    string userLocale = 'fr_FR';

    User u = new User(Alias = 'standt', Email='standarduser@testorg.com',
                      EmailEncodingKey='UTF-8', LastName='Testing', LanguageLocaleKey='en_US',
                      LocaleSidKey=userLocale, TimeZoneSidKey='America/Los_Angeles',
```
ProfileId = [SELECT Id FROM Profile WHERE Name='Standard User'].Id,
UserName='standarduser' + DateTime.now().getTime() + '@testorg.com');

System.runAs(u) {
    List<String> shoppingList = new List<String> {
        'épaule désosé Agneau',
        'Juice',
        'à la mélasse Galette 5 kg',
        'Bread',
        'Grocery'
    };

    // Default sort
    shoppingList.sort();
    Assert.areEqual('Bread', shoppingList[0]);

    // Sort based on user Locale
    Collator myCollator = Collator.getInstance();
    shoppingList.sort(myCollator);
    Assert.areEqual('à la mélasse Galette 5 kg', shoppingList[0]);
    Assert.areEqual('Bread', shoppingList[1]);
    Assert.areEqual('épaule désosé Agneau', shoppingList[2]);
    Assert.areEqual('Grocery', shoppingList[3]);
    Assert.areEqual('Juice', shoppingList[4]);
}

IN THIS SECTION:
Collator Methods

Collator Methods
The following are methods for Collator.

IN THIS SECTION:
compare(source, target)
Perform string comparisons for a given locale.
getInstance()
Gets the Collator instance for the current user’s locale.

compare(source, target)
Perform string comparisons for a given locale.

Signature
public Integer compare(String source, String target)
Parameters

source
Type: String

target
Type: String

Return Value
Type: Integer

getInstance()
Gets the Collator instance for the current user’s locale.

Signature
public static System.Collator getInstance()

Return Value
Type: Collator Class

Comparable Interface
Adds sorting support for Lists that contain non-primitive types, that is, Lists of user-defined types. Your implementation must explicitly handle null inputs in the compareTo() method to avoid a null pointer exception.

Namespace
System

Usage
To add List sorting support for your Apex class, you must implement the Comparable interface with its compareTo method in your class.

To implement the Comparable interface, you must first declare a class with the implements keyword as follows:

```java
public class Employee implements Comparable {
    // Your code here
}
```

Next, your class must provide an implementation for the following method:

```java
public Integer compareTo(Object compareTo) {
    // Your code here
}
```

The implemented method must be declared as global or public.

IN THIS SECTION:
    Comparable Methods
Comparable Example Implementation

SEE ALSO:
  - List Class

Comparable Methods
The following are methods for Comparable.

IN THIS SECTION:
  - compareTo(objectToCompareTo)
    - Returns an Integer value that is the result of the comparison.

**compareTo(objectToCompareTo)**
Returns an Integer value that is the result of the comparison.

**Signature**
```
public Integer compareTo(Object objectToCompareTo)
```

**Parameters**
- **objectToCompareTo**
  - Type: Object

**Return Value**
- Type: Integer

**Usage**
The implementation of this method returns the following values:
- 0 if this instance and objectToCompareTo are equal
- > 0 if this instance is greater than objectToCompareTo
- < 0 if this instance is less than objectToCompareTo

If this object instance and objectToCompareTo are incompatible, a System.TypeException is thrown.

Comparable Example Implementation
This example implements the Comparable interface. The compareTo method in this example compares the employee of this class instance with the employee passed in the argument. The method returns an Integer value based on the comparison of the employee IDs.

```
public class Employee implements Comparable {

    public Long id;

```
public String name;
public String phone;

// Constructor
public Employee(Long i, String n, String p) {
    id = i;
    name = n;
    phone = p;
}

// Implement the compareTo() method
public Integer compareTo(Object compareTo) {
    Employee compareToEmp = (Employee)compareTo;
    if (id == compareToEmp.id) return 0;
    if (id > compareToEmp.id) return 1;
    return -1;
}

This example tests the sort order of a list of Employee objects.

@isTest
private class EmployeeSortingTest {
  @isTest
  static void test1() {
    List<Employee> empList = new List<Employee>();
    empList.add(new Employee(101,'Joe Smith', '4155551212'));
    empList.add(new Employee(101,'J. Smith', '4155551212'));
    empList.add(new Employee(25,'Caragh Smith', '4155551000'));
    empList.add(new Employee(105,'Mario Ruiz', '4155551099'));
    // Sort using the custom compareTo() method
    empList.sort();
    // Write list contents to the debug log
    System.debug(empList);
    // Verify list sort order.
    Assert.areEqual('Caragh Smith', empList[0].Name);
    Assert.areEqual('Joe Smith', empList[1].Name);
    Assert.areEqual('J. Smith', empList[2].Name);
    Assert.areEqual('Mario Ruiz', empList[3].Name);
  }
}

Comparator Interface

Implement different sort orders with the Comparator interface’s compareTo() method, and pass the Comparator as a parameter to List.sort(). Your implementation must explicitly handle null inputs in the compareTo() method to avoid a null pointer exception.

Namespace

System
Comparator Methods

The following are methods for Comparator.

Comparator Example Implementation

Use the Comparator interface to impose different kinds of sorting.

public class Employee {
    private Long id;
    private String name;
}
private Integer yearJoined;

// Constructor
public Employee(Long i, String n, Integer y) {
    id = i;
    name = n;
    yearJoined = y;
}

public String getName() { return name; }
public Integer getYear() { return yearJoined; }

// Class to compare Employees by name
public class NameCompare implements Comparator<Employee> {
    public Integer compare(Employee e1, Employee e2) {
        if(e1?.getName() == null && e2?.getName() == null) {
            return 0;
        } else if(e1?.getName() == null) {
            return -1;
        } else if(e2?.getName() == null) {
            return 1;
        }
        return e1.getName().compareTo(e2.getName());
    }
}

// Class to compare Employees by year joined
public class YearCompare implements Comparator<Employee> {
    public Integer compare(Employee e1, Employee e2) {
        // Guard against null operands for '<' or '>' operators because
        // they will always return false and produce inconsistent sorting
        Integer result;
        if(e1?.getYear() == null && e2?.getYear() == null) {
            result = 0;
        } else if(e1?.getYear() == null) {
            result = -1;
        } else if(e2?.getYear() == null) {
            result = 1;
        } else if (e1.getYear() < e2.getYear()) {
            result = -1;
        } else if (e1.getYear() > e2.getYear()) {
            result = 1;
        } else {
            result = 0;
        }
        return result;
    }
}

The following example tests the implementation:

@isTest
private class EmployeeSortingTest {
    @isTest
static void sortWithComparators() {
    List<Employee> empList = new List<Employee>();
    empList.add(new Employee(101,'Joe Smith', 2020));
    empList.add(new Employee(102,'J. Smith', 2020));
    empList.add(new Employee(25,'Caragh Smith', 2021));
    empList.add(new Employee(105,'Mario Ruiz', 2019));
    // Sort by name
    NameCompare nameCompare = new NameCompare();
    empList.sort(nameCompare);
    // Expected order: Caragh Smith, J. Smith, Joe Smith, Mario Ruiz
    Assert.areEqual('Caragh Smith', empList.get(0).getName());

    // Sort by year joined
    YearCompare yearCompare = new YearCompare();
    empList.sort(yearCompare);
    // Expected order: Mario Ruiz, J. Smith, Joe Smith, Caragh Smith
    Assert.areEqual('Mario Ruiz', empList.get(0).getName());
}

Continuation Class

Use the Continuation class to make callouts asynchronously to a SOAP or REST Web service.

Namespace

System

Example

For a code example, see Make Long-Running Callouts from a Visualforce Page.

IN THIS SECTION:

Continuation Constructors

Continuation Properties

Continuation Methods

Continuation Constructors

The following are constructors for Continuation.

IN THIS SECTION:

Continuation(timeout)

Continuation(timeout)

Continuation Class

Continuation Class

Continuation Class

Continuation Class

Continuation Class

Continuation Class

Continuation Class

Continuation Class
**Signature**

`public Continuation(Integer timeout)`

**Parameters**

`timeout`

Type: `Integer`

The timeout for this continuation in seconds.

**Continuation Properties**

The following are properties for `Continuation`.

**IN THIS SECTION:**

- `continuationMethod`
  The name of the callback method that is called after the callout response returns.
- `timeout`
  The timeout of the continuation in seconds. Maximum: 120 seconds.
- `state`
  Data that is stored in this continuation and that can be retrieved after the callout is finished and the callback method is invoked.

**continuationMethod**

The name of the callback method that is called after the callout response returns.

**Signature**

`public String continuationMethod {get; set;}`

**Property Value**

Type: `String`

**Usage**

**Note:** If the `continuationMethod` property is not set for a Continuation, the same action method that made the asynchronous callout is called again when the callout response returns.

**timeout**

The timeout of the continuation in seconds. Maximum: 120 seconds.

**Signature**

`public Integer timeout {get; set;}`
Property Value
Type: Integer

**state**
Data that is stored in this continuation and that can be retrieved after the callout is finished and the callback method is invoked.

**Signature**
```
public Object state {get; set;}
```

Property Value
Type: Object

**Example**
This example shows how to save state information for a continuation in a controller.

```
// Declare inner class to hold state info
private class StateInfo {
    String msg { get; set; }
    List<String> urls { get; set; }
    StateInfo(String msg, List<String> urls) {
        this.msg = msg;
        this.urls = urls;
    }
}

// Then in the action method, set state for the continuation
continuationInstance.state = new StateInfo('Some state data', urls);
```

**Continuation Methods**
The following are methods for Continuation.

**addHttpRequest(request)**
Adds the HTTP request for the callout that is associated with this continuation.

**getRequests()**
Returns all labels and requests that are associated with this continuation as key-value pairs.

**getResponse(requestLabel)**
Returns the response for the request that corresponds to the specified label.

**addHttpRequest(request)**
Adds the HTTP request for the callout that is associated with this continuation.
Signature

public String addHttpRequest(System.HttpRequest request)

Parameters

request
Type: HttpRequest
The HTTP request to be sent to the external service by this continuation.

Return Value

Type: String
A unique label that identifies the HTTP request that is associated with this continuation. This label is used in the map that getRequests() returns to identify individual requests in a continuation.

Usage

You can add up to three requests to a continuation.

Note: The timeout that is set in each passed-in request is ignored. Only the global timeout maximum of 120 seconds applies for a continuation.

getRequests()

Returns all labels and requests that are associated with this continuation as key-value pairs.

Signature

public Map<String, System.HttpRequest> getRequests()

Return Value

Type: Map<String, HttpRequest>
A map of all requests that are associated with this continuation. The map key is the request label, and the map value is the corresponding HTTP request.

getResponse(requestLabel)

Returns the response for the request that corresponds to the specified label.

Signature

public static HttpResponse getResponse(String requestLabel)

Parameters

requestLabel
Type: String
The request label to get the response for.
Return Value
Type: HttpResponse

Usage
The status code is returned in the HttpResponse object and can be obtained by calling getStatusCode() on the response. A status code of 200 indicates that the request was successful. Other status code values indicate the type of problem that was encountered.

Sample of Error Status Codes
When a problem occurs with the response, some possible status code values are:

- 2000: The timeout was reached, and the server didn’t get a chance to respond.
- 2001: There was a connection failure.
- 2002: Exceptions occurred.
- 2003: The response hasn’t arrived (which also means that the Apex asynchronous callout framework hasn’t resumed).
- 2004: The response size is too large (greater than 1 MB).

Cookie Class
The Cookie class lets you access cookies for your Salesforce site using Apex.

Namespace
System

Usage
Use the setCookies method of the PageReference Class to attach cookies to a page.

Important:
- Cookie names and values set in Apex are URL encoded, that is, characters such as @ are replaced with a percent sign and their hexadecimal representation.
- The setCookies method adds the prefix “apex__” to the cookie names.
- Setting a cookie’s value to null sends a cookie with an empty string value instead of setting an expired attribute.
- After you create a cookie, the properties of the cookie can’t be changed.
- Be careful when storing sensitive information in cookies. Pages are cached regardless of a cookie value. If you use a cookie value to generate dynamic content, you should disable page caching. For more information, see Configure Site Caching in Salesforce Help.

Consider the following limitations when using the Cookie class:
- The Cookie class can only be accessed using Apex that is saved using the Salesforce API version 19 and above.
- The maximum number of cookies that can be set per Salesforce Sites domain depends on your browser. Newer browsers have higher limits than older ones.
- Cookies must be less than 4K, including name and attributes.
- The maximum header size of a Visualforce page, including cookies, is 8,192 bytes.

For more information on sites, see “Salesforce Sites” in the Salesforce online help.
Example

The following example creates a class, CookieController, which is used with a Visualforce page (see markup below) to update a counter each time a user displays a page. The number of times a user goes to the page is stored in a cookie.

```java
// A Visualforce controller class that creates a cookie
// used to keep track of how often a user displays a page
public class CookieController {
    public CookieController() {
        Cookie counter = ApexPages.currentPage().getCookies().get('counter');

        // If this is the first time the user is accessing the page,
        // create a new cookie with name 'counter', an initial value of '1',
        // path 'null', maxAge '-1', and isSecure 'true'.
        if (counter == null) {
            counter = new Cookie('counter', '1', null, -1, true);
        } else {
            // If this isn't the first time the user is accessing the page
            // create a new cookie, incrementing the value of the original count by 1
            Integer count = Integer.valueOf(counter.getValue());
            counter = new Cookie('counter', String.valueOf(count + 1), null, -1, true);
        }

        // Set the new cookie for the page
        ApexPages.currentPage().setCookies(new Cookie[]{counter});
    }

    // This method is used by the Visualforce action {!count} to display the current
    // value of the number of times a user had displayed a page.
    // This value is stored in the cookie.
    public String getCount() {
        Cookie counter = ApexPages.currentPage().getCookies().get('counter');
        if (counter == null) {
            return '0';
        }
        return counter.getValue();
    }
}

// Test class for the Visualforce controller
@isTest
private class CookieControllerTest {
    // Test method for verifying the positive test case
    static testMethod void testCounter() {
        // first page view
        CookieController controller = new CookieController();
        System.assert(controller.getCount() == '1');

        // second page view
        controller = new CookieController();
        System.assert(controller.getCount() == '2');
    }
}
```
The following is the Visualforce page that uses the CookieController Apex controller above. The action {!count} calls the getCount method in the controller above.

```apex
<apex:page controller="CookieController">
    You have seen this page {!count} times
</apex:page>
```

IN THIS SECTION:
- Cookie Constructors
- Cookie Methods

### Cookie Constructors

The following are constructors for Cookie.

IN THIS SECTION:
- `Cookie(name, value, path, maxAge, isSecure)`
  Creates a new instance of the Cookie class using the specified name, value, path, age, and the secure setting.
- `Cookie(name, value, path, maxAge, isSecure, SameSite)`
  Creates a new instance of the Cookie class using the specified name, value, path, and age, and settings for security and cross-domain behavior.

### `Cookie(name, value, path, maxAge, isSecure)`

Creates a new instance of the Cookie class using the specified name, value, path, age, and the secure setting.

**Signature**

```
public Cookie(String name, String value, String path, Integer maxAge, Boolean isSecure)
```

**Parameters**

- `name`
  - Type: String
  - The cookie name. It can't be null.

- `value`
  - Type: String
  - The cookie data, such as session ID.

- `path`
  - Type: String
  - The path from where you can retrieve the cookie.

- `maxAge`
  - Type: Integer
  - A number representing how long a cookie is valid for in seconds. If set to less than zero, a session cookie is issued. If set to zero, the cookie is deleted.
isSecure
  Type: Boolean
  A value indicating whether the cookie can only be accessed through HTTPS (true) or not (false).

Cookie(name, value, path, maxAge, isSecure, SameSite)
Creates a new instance of the Cookie class using the specified name, value, path, and age, and settings for security and cross-domain behavior.

Note: Google Chrome 80 introduces a new default cookie attribute setting of SameSite, which is set to Lax. Previously, the SameSite cookie attribute defaulted to the value of None. When SameSite is set to None, cookies must be tagged with the isSecure attribute indicating that they require an encrypted HTTPS connection.

Signature
public Cookie(String name, String value, String path, Integer maxAge, Boolean isSecure, String SameSite)

Parameters
name
  Type: String
  The cookie name. It can’t be null.

value
  Type: String
  The cookie data, such as session ID.

path
  Type: String
  The path from where you can retrieve the cookie.

maxAge
  Type: Integer
  A number representing how long a cookie is valid for in seconds. If set to less than zero, a session cookie is issued. If set to zero, the cookie is deleted.

isSecure
  Type: Boolean
  A value indicating whether the cookie can only be accessed through HTTPS (true) or not (false).

SameSite
  Type: String
The SameSite attribute on a cookie controls its cross-domain behavior. The valid values are None, Lax, and Strict. After the Chrome 80 release, a cookie with a SameSite value of None must also be marked secure by setting a value of None; Secure.

SEE ALSO:

Salesforce Spring '20 Release Notes: Prepare for Google Chrome's Changes in SameSite Cookie Behavior That Can Break Salesforce Integrations
Chrome Platform Status: Reject insecure SameSite=None cookies

Cookie Methods

The following are methods for Cookie. All are instance methods.

IN THIS SECTION:

getDomain()
Returns the name of the server making the request.

getMaxAge()
Returns a number representing how long the cookie is valid for, in seconds. If set to < 0, a session cookie is issued. If set to 0, the cookie is deleted.

getName()
Returns the name of the cookie. Can't be null.

getPath()
Returns the path from which you can retrieve the cookie. If null or blank, the location is set to root, or "/".

getSameSite()
Returns the value for the SameSite attribute of the cookie.

getValue()
Returns the data captured in the cookie, such as Session ID.

isSecure()
Returns true if the cookie can only be accessed through HTTPS, otherwise returns false.

getDomain()
Returns the name of the server making the request.

Signature

public String getDomain()
**getMaxAge()**
Returns a number representing how long the cookie is valid for, in seconds. If set to $< 0$, a session cookie is issued. If set to 0, the cookie is deleted.

**Signature**
```
public Integer getMaxAge()
```

**Return Value**
Type: Integer

**getName()**
Returns the name of the cookie. Can't be null.

**Signature**
```
public String getName()
```

**Return Value**
Type: String

**getPath()**
Returns the path from which you can retrieve the cookie. If null or blank, the location is set to root, or "/".

**Signature**
```
public String getPath()
```

**Return Value**
Type: String

**getSameSite()**
Returns the value for the SameSite attribute of the cookie.

**Signature**
```
public String getSameSite()
```

**Return Value**
Type: String

SEE ALSO:
web.dev: SameSite Cookies Explained
getValue()  
Returns the data captured in the cookie, such as Session ID.

Signature  
public String getValue()  

Return Value  
Type: String

isSecure()  
Returns true if the cookie can only be accessed through HTTPS, otherwise returns false.

Signature  
public Boolean isSecure()  

Return Value  
Type: Boolean

Crypto Class  
Provides methods for creating digests, message authentication codes, and signatures, as well as encrypting and decrypting information.

Namespace  
System

Usage  
The methods in the Crypto class can be used for securing content in Lightning Platform, or for integrating with external services such as Google or Amazon WebServices (AWS).

Encrypt and Decrypt Exceptions  
The following exceptions can be thrown for these methods:

- decrypt  
- encrypt  
- decryptWithManagedIV  
- encryptWithManagedIV
<table>
<thead>
<tr>
<th>Exception</th>
<th>Message</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>InvalidParameterValue</td>
<td>Unable to parse initialization vector from encrypted data.</td>
<td>Thrown if you’re using managed initialization vectors, and the cipher text is less than 16 bytes.</td>
</tr>
<tr>
<td>InvalidParameterValue</td>
<td>Invalid algorithm <code>algoName</code>. Must be AES128, AES192, AES256, AES384, or AES512.</td>
<td>Thrown if the algorithm name isn’t one of the valid values.</td>
</tr>
<tr>
<td>InvalidParameterValue</td>
<td>Invalid private key. Must be <code>size</code> bytes.</td>
<td>Thrown if size of the private key doesn’t match the specified algorithm.</td>
</tr>
<tr>
<td>InvalidParameterValue</td>
<td>Invalid initialization vector. Must be 16 bytes.</td>
<td>Thrown if the initialization vector isn’t 16 bytes.</td>
</tr>
<tr>
<td>InvalidParameterValue</td>
<td>Invalid data. Input data is <code>size</code> bytes, which exceeds the limit of 1048576 bytes.</td>
<td>Thrown if the data is greater than 1 MB. For decryption, 1048608 bytes are allowed for the initialization vector header, plus any additional padding the encryption added to align to block size.</td>
</tr>
<tr>
<td>NullPointer.Exception</td>
<td>Argument cannot be null.</td>
<td>Thrown if one of the required method arguments is null.</td>
</tr>
<tr>
<td>SecurityException</td>
<td>Given final block not properly padded.</td>
<td>Thrown if the data isn’t properly block-aligned or similar issues occur during encryption or decryption.</td>
</tr>
<tr>
<td>SecurityException</td>
<td>Message Varies</td>
<td>Thrown if something goes wrong during either encryption or decryption.</td>
</tr>
</tbody>
</table>

**Crypto Methods**

The following are methods for `Crypto`. All methods are static.

**IN THIS SECTION:**

- `decrypt(algorithmName, privateKey, initializationVector, cipherText)`
  - Decrypts the Blob `cipherText` using the specified algorithm, private key, and initialization vector. Use this method to decrypt blobs encrypted using a third party application or the `encrypt` method.

- `decryptWithManagedIV(algorithmName, privateKey, IVAndCipherText)`
  - Decrypts the Blob `IVAndCipherText` using the specified algorithm and private key. Use this method to decrypt blobs encrypted using a third party application or the `encryptWithManagedIV` method.

- `encrypt(algorithmName, privateKey, initializationVector, clearText)`
  - Encrypts the Blob `clearText` using the specified algorithm, private key and initialization vector. Use this method when you want to specify your own initialization vector.

- `encryptWithManagedIV(algorithmName, privateKey, clearText)`
  - Encrypts the Blob `clearText` using the specified algorithm and private key. Use this method when you want Salesforce to generate the initialization vector for you.
generateAesKey(size)
Generates an Advanced Encryption Standard (AES) key.
generateDigest(algorithmName, input)
Computes a secure, one-way hash digest based on the supplied input string and algorithm name.
generateMac(algorithmName, input, privateKey)
Computes a message authentication code (MAC) for the input string, using the private key and the specified algorithm.
getRandomInteger()
Returns a random Integer.
getRandomLong()
Returns a random Long.
sign(algorithmName, input, privateKey)
Computes a unique digital signature for the input string, using the specified algorithm and the supplied private key.
signWithCertificate(algorithmName, input, certDevName)
Computes a unique digital signature for the input string, using the specified algorithm and the supplied certificate and key pair.
signXML(algorithmName, node, idAttributeName, certDevName)
Envelops the signature into an XML document.
signXML(algorithmName, node, idAttributeName, certDevName, refChild)
Inserts the signature envelope before the specified child node.
verify(String algorithmName, Blob data, Blob signature, Blob publicKey)
Verifies the digital signature for the Blob data using the specified algorithm and the supplied public key. Use this method to verify a Blob signed by a digital signature created using a third-party application or the sign method.
verify(String algorithmName, Blob data, Blob signature, String certDevName)
Verifies the digital signature for the Blob data using the specified algorithm and the public key associated with the certDevName. Use this method to verify a Blob signed by a digital signature created using a third-party application or the sign method.
verifyHmac(String algorithmName, Blob input, Blob privateKey, Blob macToVerify)
Verifies the HMAC signature for Blob data using the specified algorithm, input data, privateKey, and the mac. Use this method to verify a Blob signed by a digital signature created using a third-party application or the sign method.

decrypt(algorithmName, privateKey, initializationVector, cipherText)
Decrypts the Blob cipherText using the specified algorithm, private key, and initialization vector. Use this method to decrypt blobs encrypted using a third party application or the encrypt method.

Signature

public static Blob decrypt(String algorithmName, Blob privateKey, Blob initializationVector, Blob cipherText)

Parameters

algorithmName
  Type: String
privateKey
  Type: Blob
**initializationVector**
Type: Blob

**cipherText**
Type: Blob

**Return Value**
Type: Blob

**Usage**
Valid values for *algorithmName* are:
- AES128
- AES192
- AES256

These algorithms are all industry standard Advanced Encryption Standard (AES) algorithms with different size keys. They use cipher block chaining (CBC) and PKCS7 padding.

The length of *privateKey* must match the specified algorithm: 128 bits, 192 bits, or 256 bits, which is 16 bytes, 24 bytes, or 32 bytes, respectively. You can use a third-party application or the *generateAesKey* method to generate this key for you.

The initialization vector must be 128 bits (16 bytes.)

**Example**

```java
Blob exampleIv = Blob.valueOf('Example of IV123');
Blob key = Crypto.generateAesKey(128);
Blob data = Blob.valueOf('Data to be encrypted');
Blob encrypted = Crypto.encrypt('AES128', key, exampleIv, data);

Blob decrypted = Crypto.decrypt('AES128', key, exampleIv, encrypted);
String decryptedString = decrypted.toString();
System.assertEquals('Data to be encrypted', decryptedString);
```

**decryptWithManagedIV(algorithmName, privateKey, IVAndCipherText)**

Decrypts the Blob *IVAndCipherText* using the specified algorithm and private key. Use this method to decrypt blobs encrypted using a third party application or the *encryptWithManagedIV* method.

**Signature**

```java
public static Blob decryptWithManagedIV(String algorithmName, Blob privateKey, Blob IVAndCipherText)
```

**Parameters**

*algorithmName*
Type: String
**privateKey**
Type: Blob

**IVAndCipherText**
Type: Blob
The first 128 bits (16 bytes) of IVAndCipherText must contain the initialization vector.

**Return Value**
Type: Blob

**Usage**
Valid values for `algorithmName` are:
- AES128
- AES192
- AES256

These algorithms are all industry standard Advanced Encryption Standard (AES) algorithms with different size keys. They use cipher block chaining (CBC) and PKCS7 padding.

The length of privateKey must match the specified algorithm: 128 bits, 192 bits, or 256 bits, which is 16 bytes, 24 bytes, or 32 bytes, respectively. You can use a third-party application or the `generateAesKey` method to generate this key for you.

**Example**

```java
Blob key = Crypto.generateAesKey(128);
Blob data = Blob.valueOf('Data to be encrypted');
Blob encrypted = Crypto.encryptWithManagedIV('AES128', key, data);

Blob decrypted = Crypto.decryptWithManagedIV('AES128', key, encrypted);
String decryptedString = decrypted.toString();
System.assertEquals('Data to be encrypted', decryptedString);
```

**encrypt(algorithmName, privateKey, initializationVector, clearText)**
Encrypts the Blob `clearText` using the specified algorithm, private key and initialization vector. Use this method when you want to specify your own initialization vector.

**Signature**

```java
public static Blob encrypt(String algorithmName, Blob privateKey, Blob initializationVector, Blob clearText)
```

**Parameters**

- `algorithmName`
  Type: String
- `privateKey`
  Type: Blob
**Usage**

The initialization vector must be 128 bits (16 bytes.) Use either a third-party application or the `decrypt` method to decrypt blobs encrypted using this method. Use the `encryptWithManagedIV` method if you want Salesforce to generate the initialization vector for you. It is stored as the first 128 bits (16 bytes) of the encrypted Blob.

Valid values for `algorithmName` are:

- AES128
- AES192
- AES256

These algorithms are all industry standard Advanced Encryption Standard (AES) algorithms with different size keys. They use cipher block chaining (CBC) and PKCS7 padding.

The length of `privateKey` must match the specified algorithm: 128 bits, 192 bits, or 256 bits, which is 16 bytes, 24 bytes, or 32 bytes, respectively. You can use a third-party application or the `generateAesKey` method to generate this key for you.

**Example**

```java
Blob exampleIv = Blob.valueOf('Example of IV123');
Blob key = Crypto.generateAesKey(128);
Blob data = Blob.valueOf('Data to be encrypted');
Blob encrypted = Crypto.encrypt('AES128', key, exampleIv, data);

Blob decrypted = Crypto.decrypt('AES128', key, exampleIv, encrypted);
String decryptedString = decrypted.toString();
System.assertEquals('Data to be encrypted', decryptedString);
```

**encryptWithManagedIV(algorithmName, privateKey, clearText)**

Encrypts the Blob `clearText` using the specified algorithm and private key. Use this method when you want Salesforce to generate the initialization vector for you.

**Signature**

```java
public static Blob encryptWithManagedIV(String algorithmName, Blob privateKey, Blob clearText)
```

**Parameters**

- `algorithmName`
  Type: `String`
privateKey
  Type: Blob
clearText
  Type: Blob

Return Value
Type: Blob

Usage
The initialization vector is stored as the first 128 bits (16 bytes) of the encrypted Blob. Use either third-party applications or the decryptWithManagedIV method to decrypt blobs encrypted with this method. Use the encrypt method if you want to generate your own initialization vector.

Valid values for algorithmName are:

- AES128
- AES192
- AES256

These algorithms are all industry standard Advanced Encryption Standard (AES) algorithms with different size keys. They use cipher block chaining (CBC) and PKCS7 padding.

The length of privateKey must match the specified algorithm: 128 bits, 192 bits, or 256 bits, which is 16 bytes, 24 bytes, or 32 bytes, respectively. You can use a third-party application or the generateAesKey method to generate this key for you.

Example

```java
Blob key = Crypto.generateAesKey(128);
Blob data = Blob.valueOf('Data to be encrypted');
Blob encrypted = Crypto.encryptWithManagedIV('AES128', key, data);

Blob decrypted = Crypto.decryptWithManagedIV('AES128', key, encrypted);
String decryptedString = decrypted.toString();
System.assertEquals('Data to be encrypted', decryptedString);
```

generateAesKey(size)
Generates an Advanced Encryption Standard (AES) key.

Signature
public static Blob generateAesKey(Integer size)

Parameters
size
  Type: Integer
  The key's size in bits. Valid values are:
  - 128
Return Value
Type: Blob

Example

```java
Blob key = Crypto.generateAesKey(128);
```

generateDigest(algorithmName, input)
Computes a secure, one-way hash digest based on the supplied input string and algorithm name.

Signature

```java
public static Blob generateDigest(String algorithmName, Blob input)
```

Parameters

- `algorithmName`
  Type: String
  Valid values for `algorithmName` are:
  - MD5
  - SHA1
  - SHA3-256
  - SHA3-384
  - SHA3-512
  - SHA-256
  - SHA-512

- `input`
  Type: Blob

Return Value
Type: Blob

Example

```java
Blob targetBlob = Blob.valueOf('ExampleMD5String');
Blob hash = Crypto.generateDigest('MD5', targetBlob);
```

generateMac(algorithmName, input, privateKey)
Computes a message authentication code (MAC) for the input string, using the private key and the specified algorithm.
Signature

```java
public static Blob generateMac(String algorithmName, Blob input, Blob privateKey)
```

Parameters

- `algorithmName`
  - Type: `String`
  - The valid values for `algorithmName` are:
    - hmacMD5
    - hmacSHA1
    - hmacSHA256
    - hmacSHA512

- `input`
  - Type: `Blob`

- `privateKey`
  - Type: `Blob`
  - The value of `privateKey` does not need to be in decoded form. The value cannot exceed 4 KB.

Return Value

- Type: `Blob`

Example

```java
String salt = String.valueOf(Crypto.getRandomInteger());
String key = 'key';
Blob data = crypto.generateMac('HmacSHA256',
   Blob.valueOf(salt), Blob.valueOf(key));
```

`getRandomInteger()`

Returns a random Integer.

Signature

```java
public static Integer getRandomInteger()
```

Return Value

- Type: `Integer`

Example

```java
Integer randomInt = Crypto.getRandomInteger();
```
**getRandomLong()**

Returns a random Long.

**Signature**

```java
public static Long getRandomLong()
```

**Return Value**

Type: Long

**Example**

```java
Long randomLong = Crypto.getRandomLong();
```

**sign(algorithmName, input, privateKey)**

Computes a unique digital signature for the input string, using the specified algorithm and the supplied private key.

**Signature**

```java
public static Blob sign(String algorithmName, Blob input, Blob privateKey)
```

**Parameters**

- **algorithmName**
  - Type: String
  - The algorithm name. The valid values for `algorithmName` are RSA, RSA-SHA1, RSA-SHA256, RSA-SHA384, RSA-SHA512, ECDSA-SHA256, ECDSA-SHA384, and ECDSA-SHA512.
  - RSA-SHA1 is an RSA signature (with an asymmetric key pair) of an SHA1 hash.
  - RSA-SHA256 is an RSA signature of an SHA256 hash.
  - RSA-SHA384 is an RSA signature of an SHA384 hash.
  - RSA-SHA512 is an RSA signature of an SHA512 hash.
  - RSA is the same as RSA-SHA1.
  - ECDSA-SHA256 is an ECDSA signature of an SHA256 hash.
  - ECDSA-SHA384 is an ECDSA signature of an SHA384 hash.
  - ECDSA-SHA512 is an ECDSA signature of an SHA512 hash.

- **input**
  - Type: Blob
  - The data to sign.

- **privateKey**
  - Type: Blob
  - The value of `privateKey` must be decoded using the `EncodingUtilBase64Decode` method, and should be in RSA's PKCS #8 (1.2) Private-Key Information Syntax Standard form. The value cannot exceed 4 KB.
Return Value

Type: Blob

Example

The following snippet shows how to call the `sign` method.

```java
String algorithmName = 'RSA';
String key = '';
Blob privateKey = EncodingUtil.base64Decode(key);
Blob input = Blob.valueOf('12345qwerty');
Crypto.sign(algorithmName, input, privateKey);
```

`signWithCertificate(algorithmName, input, certDevName)`

Computes a unique digital signature for the input string, using the specified algorithm and the supplied certificate and key pair.

**Signature**

```java
public static Blob signWithCertificate(String algorithmName, Blob input, String certDevName)
```

**Parameters**

- `algorithmName`
  - Type: String
  - The algorithm name. The valid values for `algorithmName` are `RSA`, `RSA-SHA1`, `RSA-SHA256`, `RSA-SHA384`, `RSA-SHA512`, `ECDSA-SHA256`, `ECDSA-SHA384`, and `ECDSA-SHA512`.
  - `RSA-SHA1` is an RSA signature (with an asymmetric key pair) of an SHA1 hash.
  - `RSA-SHA256` is an RSA signature of an SHA256 hash.
  - `RSA-SHA384` is an RSA signature of an SHA384 hash.
  - `RSA-SHA512` is an RSA signature of an SHA512 hash.
  - `RSA` is the same as `RSA-SHA1`.
  - `ECDSA-SHA256` is an ECDSA signature of an SHA256 hash.
  - `ECDSA-SHA384` is an ECDSA signature of an SHA384 hash.
  - `ECDSA-SHA512` is an ECDSA signature of an SHA512 hash.

- `input`
  - Type: Blob
  - The data to sign.

- `certDevName`
  - Type: String
  - The `Unique Name` for a certificate stored in the Salesforce org’s Certificate and Key Management page to use for signing.

To access the Certificate and Key Management page from Setup, enter `Certificate and Key Management` in the Quick Find box, then select `Certificate and Key Management`.
Return Value

Type: Blob

Example

The following snippet is an example of the method for signing the content referenced by `data`.

```java
Blob data = Blob.valueOf('12345qwerty');
System.Crypto.signWithCertificate('RSA-SHA512', data, 'signingCert');
```

`signXML(algorithmName, node, idAttributeName, certDevName)`

Envelops the signature into an XML document.

Signature

```java
public Void signXML(String algorithmName, Dom.XmlNode node, String idAttributeName, String certDevName)
```

Parameters

`algorithmName`

Type: String

The RSA encryption algorithm name. Valid names are RSA, RSA-SHA1, RSA-SHA256, RSA-SHA384, RSA-SHA512, ECDSA-SHA256, ECDSA-SHA384, and ECDSA-SHA512.

- RSA-SHA1 is an RSA signature (with an asymmetric key pair) of an SHA1 hash.
- RSA-SHA256 is an RSA signature of an SHA256 hash.
- RSA-SHA384 is an RSA signature of an SHA384 hash.
- RSA-SHA512 is an RSA signature of an SHA512 hash.
- RSA is the same as RSA-SHA1.
- ECDSA-SHA256 is an ECDSA signature of an SHA256 hash.
- ECDSA-SHA384 is an ECDSA signature of an SHA384 hash.
- ECDSA-SHA512 is an ECDSA signature of an SHA512 hash.

`node`

Type: Dom.XmlNode

The XML node to sign and insert the signature into.

`idAttributeName`

Type: String

The full name (including the namespace) of the attribute on the node (XmlNode) to use as the reference ID. If `null`, this method uses the ID attribute on the node. If there is no ID attribute, Salesforce generates a new ID and adds it to the node.

`certDevName`

Type: String

The unique name for a certificate stored in the Salesforce org’s Certificate and Key Management page to use for signing.
To access the Certificate and Key Management page from Setup, enter Certificate and Key Management in the Quick Find box, then select Certificate and Key Management.

Return Value
Type: void

Example
The following is an example declaration and initialization.

```java
doc.load(...);
System.Crypto.signXml('RSA-SHA512', doc.getRootElement(), null, 'signingCert');
return doc.toXmlString();
```

signXML(algorithmName, node, idAttributeName, certDevName, refChild)
Inserts the signature envelope before the specified child node.

Signature

```java
public static void signXml(String algorithmName, Dom.XmlNode node, String idAttributeName, String certDevName, Dom.XmlNode refChild)
```

Parameters

**algorithmName**
Type: String
The RSA encryption algorithm name. Valid names are RSA, RSA-SHA1, RSA-SHA256, RSA-SHA384, RSA-SHA512, ECDSA-SHA256, ECDSA-SHA384, and ECDSA-SHA512.

RSA-SHA1 is an RSA signature (with an asymmetric key pair) of an SHA1 hash.

RSA-SHA256 is an RSA signature of an SHA256 hash.

RSA-SHA384 is an RSA signature of an SHA384 hash.

RSA-SHA512 is an RSA signature of an SHA512 hash.

RSA is the same as RSA-SHA1.

ECDSA-SHA256 is an ECDSA signature of an SHA256 hash.

ECDSA-SHA384 is an ECDSA signature of an SHA384 hash.

ECDSA-SHA512 is an ECDSA signature of an SHA512 hash.

**node**
Type: Dom.XmlNode
The XML node to sign and insert the signature into.

**idAttributeName**
Type: String
The full name (including the namespace) of the attribute on the node (XmlNode) to use as the reference ID. If null, this method uses the ID attribute on the node. If there is no ID attribute, Salesforce generates a new ID and adds it to the node.
certDevName
  Type: String
  The unique name for a certificate stored in the Salesforce org’s Certificate and Key Management page to use for signing.
  To access the Certificate and Key Management page from Setup, enter Certificate and Key Management in the Quick Find box, then select Certificate and Key Management.

refChild
  Dom.XmlNode
  The XML node before which to insert the signature. If refChild is null, the signature is added at the end.

Return Value
  Type: Void

verify(String algorithmName, Blob data, Blob signature, Blob publicKey)
  Verifies the digital signature for the Blob data using the specified algorithm and the supplied public key. Use this method to verify a Blob signed by a digital signature created using a third-party application or the sign method.

Signature
  public static Blob verify(String algorithmName, Blob data, Blob signature, Blob publicKey)

Parameters
  algorithmName
    Type: String
    The algorithm name. The valid values for algorithmName are RSA, RSA-SHA1, RSA-SHA256, RSA-SHA384, RSA-SHA512, ECDSA-SHA256, ECDSA-SHA384, and ECDSA-SHA512.
    RSA-SHA1 is an RSA signature (with an asymmetric key pair) of an SHA1 hash.
    RSA-SHA256 is an RSA signature of an SHA256 hash.
    RSA-SHA384 is an RSA signature of an SHA384 hash.
    RSA-SHA512 is an RSA signature of an SHA512 hash.
    RSA is the same as RSA-SHA1.
    ECDSA-SHA256 is an ECDSA signature of an SHA256 hash.
    ECDSA-SHA384 is an ECDSA signature of an SHA384 hash.
    ECDSA-SHA512 is an ECDSA signature of an SHA512 hash.

data
    Type: Blob
    The data to sign.

signature
    Type:
    Blob
The RSA signature.

**publicKey**

Type: Blob

The value of *publicKey* must be decoded using the `EncodingUtil.base64Decode` method, and be in X.509 standard.

### Return Value

Type: Blob

### Example

```java
String algorithmName = 'RSA';
String privateKey = '';
String publicKey = '';  // Ensure to decode the publicKey
Blob privateKey = EncodingUtil.base64Decode(privateKey);
Blob publicKey = EncodingUtil.base64Decode(publicKey);
Blob input = Blob.valueOf('12345qwerty');
Blob signature = Crypto.sign(algorithmName, input, privateKey);
Boolean verified = Crypto.verify(algorithmName, input, signature, publicKey);
```

### verify(String algorithmName, Blob data, Blob signature, String certDevName)

Verifies the digital signature for the Blob *data* using the specified algorithm and the public key associated with the `certDevName`. Use this method to verify a Blob signed by a digital signature created using a third-party application or the `sign` method.

### Signature

```java
public static Blob verify(String algorithmName, Blob data, Blob signature, String certDevName)
```

### Parameters

- **algorithmName**
  Type: String
  The algorithm name. The valid values for *algorithmName* are RSA, RSA-SHA1, RSA-SHA256, RSA-SHA384, RSA-SHA512, ECDSA-SHA256, ECDSA-SHA384, and ECDSA-SHA512.
  - RSA-SHA1 is an RSA signature (with an asymmetric key pair) of an SHA1 hash.
  - RSA-SHA256 is an RSA signature of an SHA256 hash.
  - RSA-SHA384 is an RSA signature of an SHA384 hash.
  - RSA-SHA512 is an RSA signature of an SHA512 hash.
  - RSA is the same as RSA-SHA1.
  - ECDSA-SHA256 is an ECDSA signature of an SHA256 hash.
  - ECDSA-SHA384 is an ECDSA signature of an SHA384 hash.
  - ECDSA-SHA512 is an ECDSA signature of an SHA512 hash.

- **data**
  Type: Blob
The data to sign.

**signature**

Type: Blob

The RSA signature.

**certDevName**

Type: String

The Unique Name for a certificate stored in the Salesforce organization’s Certificate and Key Management page to use for signing.

To access the Certificate and Key Management page from Setup, enter Certificate and Key Management in the Quick Find box, then select Certificate and Key Management.

**Return Value**

Type: Blob

**Example**

```java
Blob data = Blob.valueOf('12345qwerty');
Blob signature = Crypto.signWithCertificate('RSA-SHA256', data, 'signingCert');
Boolean verified = Crypto.verify('RSA-SHA512', data, signature, 'signingCert');
```

**verifyHMac(String algorithmName, Blob input, Blob privateKey, Blob macToVerify)**

Verifies the HMAC signature for Blob `data` using the specified algorithm, input data, privateKey, and the mac. Use this method to verify a Blob signed by a digital signature created using a third-party application or the sign method.

**Signature**

```java
public static Blob verifyHMac(String algorithmName, Blob input, Blob privateKey, Blob macToVerify)
```

**Parameters**

- **algorithmName**
  Type: String
  The valid values for `algorithmName` are:
  - hmacMD5
  - hmacSHA1
  - hmacSHA256
  - hmacSHA512

- **data**
  Type: Blob
  The data to sign.

- **privateKey**
  Type: Blob
The value of `privateKey` does not need to be in decoded form. The value cannot exceed 4 KB.

`hmacToVerify`
- Type: `Blob`

The value of the mac must be verified against the provided `privateKey`, data, and algorithm.

**Return Value**
- Type: `Boolean`

**Example**
```
String salt = String.valueOf(Crypto.getRandomInteger());
String key = 'key';
Blob mac = Crypto.generateMac('HmacSHA256', Blob.valueOf(salt), Blob.valueOf(key));
Boolean verified = Crypto.verifyHMac('HmacSHA256', Blob.valueOf(salt), Blob.valueOf(key),
                                    mac);
```

**Custom Metadata Type Methods**
Custom metadata types are customizable, deployable, packageable, and upgradeable application metadata. All custom metadata is exposed in the application cache, which allows access without repeated queries to the database. The metadata is then available for formula fields, validation rules, flows, Apex, and SOAP API. All methods are static.

**Usage**
Custom metadata types methods are instance type methods and are called by and operate on a specific instance of a custom metadata type.

**Custom Metadata Types Example**
The following example uses the `getAll()` method. The custom metadata type named Games has a field called `GameType__c`. This example determines if the field value of the first record is equal to the string `PC`.

```
List<Games__mdt> mcs = Games__mdt.getAll().values();
boolean textField = null;
if (mcs[0].GameType__c == 'PC') {
    textField = true;
}
```

**IN THIS SECTION:**
- `getAll()`: Returns a map containing custom metadata records for the specific custom metadata type. The map keys are the record DeveloperNames and the map values are the record sObjects.
- `getInstance(recordId)`: Returns a single custom metadata type record sObject for a specified record ID.
getInstance(developerName)
Returns a single custom metadata type record sObject for a specified developerName field of the custom metadata type object.

getInstance(qualifiedApiName)
Returns a single custom metadata type record sObject for a qualified API name.

getAll()
Returns a map containing custom metadata records for the specific custom metadata type. The map keys are the record DeveloperNames and the map values are the record sObjects.

Signature
public Map<String, CustomMetadataType__mdt> getAll()

Return Value
Type: Map<String, CustomMetadataType__mdt>

Usage
If no records are defined for the type, this method returns an empty map. To iterate over the list of custom metadata type record sObjects, use getAll().values(). Only the first 255 characters are returned for any field in a custom metadata type record, so longer text fields get truncated. If you want all the field data from a custom metadata type record, use a SOQL query.

Example
This sample returns a map of all the records for a custom metadata type named Games__mdt.

Map<String, Games__mdt> mcs = Games__mdt.getAll();

gGetInstance(recordId)
Returns a single custom metadata type record sObject for a specified record ID.

Signature
public CustomMetadataType__mdt getInstance(recordId)

Parameters
recordId
Type: String

Return Value
Type: CustomMetadataType__mdt
Usage
Use this method to explicitly retrieve custom metadata type information at the user level. Only the first 255 characters of any field in a custom metadata type record are returned. Therefore, fields such as long text fields can be truncated. If you want all the field data from a custom metadata type record, use a SOQL query.

Example
This sample returns a single record sObject for the custom metadata type named Games_mdt with recordID specified as m000000000000000001.

Games__mdt mc = Games__mdt.getInstance('m000000000000000001');

getInstance(developerName)
Returns a single custom metadata type record sObject for a specified developerName field of the custom metadata type object.

Signature
public CustomMetadataType__mdt getInstance(String developerName)

Parameters
developerName
Type: String

Return Value
Type: CustomMetadataType__mdt

Usage
Use this method to return a single custom metadata type record for the specified developerName. The developerName is the unique name of the custom metadata type object in the API. Only the first 255 characters of any field in a custom metadata type record are returned. Therefore, fields such as long text fields can be truncated. If you want all the field data from a custom metadata type record, use a SOQL query.

Example
Returns a single record sObject for the custom metadata type named Games_mdt with developerName specified as FirstRecord.

Games__mdt mc = Games__mdt.getInstance('FirstRecord');

getInstance(qualifiedApiName)
Returns a single custom metadata type record sObject for a qualified API name.
Signature

public CustomMetadataType__mdt getInstance(String qualifiedApiName)

Parameters

qualifiedApiName

Type: String

Return Value

Type: CustomMetadataType__mdt

Usage

Use this method to return a single custom metadata type record for the specified qualifiedApiName. The qualifiedApiName is a concatenation of the namespace prefix and developerName, and has this format: namespacePrefix__developerName. The developerName is the unique name of the custom metadata type object in the API. Only the first 255 characters of any field in a custom metadata type record are returned. Therefore, fields such as long text fields can be truncated. If you want all the field data from a custom metadata type record, use a SOQL query.

Example

This sample returns a single record sObject for the custom metadata type named Games__mdt with qualified API name specified as MyNamespace__FirstRecord.

    Games__mdt mc = Games__mdt.getInstance('MyNamespace__FirstRecord');

Custom Settings Methods

Custom settings are similar to custom objects and enable application developers to create custom sets of data, as well as create and associate custom data for an organization, profile, or specific user. All custom settings data is exposed in the application cache, which enables efficient access without the cost of repeated queries to the database. This data is then available for formula fields, validation rules, flows, Apex, and the SOAP API.

Usage

Custom settings methods are all instance methods, that is, they are called by and operate on a specific instance of a custom setting. There are two types of custom settings: hierarchy and list. There are two types of methods: methods that work with list custom settings, and methods that work with hierarchy custom settings.

Note: All custom settings data is exposed in the application cache, which enables efficient access without the cost of repeated queries to the database. However, querying custom settings data using Standard Object Query Language (SOQL) doesn't use the application cache and is similar to querying a custom object. To benefit from caching, use other methods for accessing custom settings data such as the Apex Custom Settings methods.

For more information on creating custom settings in the Salesforce user interface, see “Create Custom Settings” in the Salesforce online help.
Custom Setting Examples

The following example uses a list custom setting called Games. The Games setting has a field called GameType. This example determines if the value of the first data set is equal to the string 'PC'.

```java
List<Games__C> mcs = Games__c.getall().values();
boolean textField = null;
if (mcs[0].GameType__c == 'PC') {
    textField = true;
}
System.assertEquals(textField, true);
```

The following example uses a custom setting called Foundation_Countries. This example demonstrates that the getValues and getInstance methods return identical values.

```java
Foundation_Countries__c myCS1 = Foundation_Countries__c.getValues('United States');
String myCCVal = myCS1.Country_code__c;
Foundation_Countries__c myCS2 = Foundation_Countries__c.getInstance('United States');
String myCCInst = myCS2.Country_code__c;
System.assertEquals(myCCInst, myCCVal);
```

Hierarchy Custom Setting Examples

In the following example, the hierarchy custom setting GamesSupport has a field called Corporate_number. The code returns the value for the profile specified with pid.

```java
GamesSupport__c mhc = GamesSupport__c.getInstance(pid);
string mPhone = mhc.Corporate_number__c;
```

The example is identical if you choose to use the getValues method.

The following example shows how to use hierarchy custom settings methods. For getInstance, the example shows how field values that aren't set for a specific user or profile are returned from fields defined at the next lowest level in the hierarchy. The example also shows how to use getOrgDefaults.

Finally, the example demonstrates how getValues returns fields in the custom setting record only for the specific user or profile, and doesn't merge values from other levels of the hierarchy. Instead, getValues returns null for any fields that aren't set. This example uses a hierarchy custom setting called Hierarchy. Hierarchy has two fields: OverrideMe and DontOverrideMe. In addition, a user named Robert has a System Administrator profile. The organization, profile, and user settings for this example are as follows:

**Organization settings**
- OverrideMe: Hello
- DontOverrideMe: World

**Profile settings**
- OverrideMe: Goodbye
- DontOverrideMe is not set.

**User settings**
- OverrideMe: Fluffy
- DontOverrideMe is not set.
The following example demonstrates the result of the getInstance method when Robert calls it in his organization:

```java
Hierarchy__c CS = Hierarchy__c.getInstance();
System.Assert(CS.OverrideMe__c == 'Fluffy');
System.assert(CS.DontOverrideMe__c == 'World');
```

If Robert passes his user ID specified by RobertId to getInstance, the results are the same. The identical results are because the lowest level of data in the custom setting is specified at the user level.

```java
Hierarchy__c CS = Hierarchy__c.getInstance(RobertId);
System.Assert(CS.OverrideMe__c == 'Fluffy');
System.assert(CS.DontOverrideMe__c == 'World');
```

If Robert passes the System Administrator profile ID specified by SysAdminID to getInstance, the result is different. The data specified for the profile is returned:

```java
Hierarchy__c CS = Hierarchy__c.getInstance(SysAdminID);
System.Assert(CS.OverrideMe__c == 'Goodbye');
System.assert(CS.DontOverrideMe__c == 'World');
```

When Robert tries to return the data set for the organization using getOrgDefaults, the result is:

```java
Hierarchy__c CS = Hierarchy__c.getOrgDefaults();
System.Assert(CS.OverrideMe__c == 'Hello');
System.assert(CS.DontOverrideMe__c == 'World');
```

By using the getValues method, Robert can get the hierarchy custom setting values specific to his user and profile settings. For example, if Robert passes his user ID RobertId to getValues, the result is:

```java
Hierarchy__c CS = Hierarchy__c.getValues(RobertId);
System.Assert(CS.OverrideMe__c == 'Fluffy');
// Note how this value is null, because you are returning
// data specific for the user
System.assert(CS.DontOverrideMe__c == null);
```

If Robert passes his System Administrator profile ID SysAdminID to getValues, the result is:

```java
Hierarchy__c CS = Hierarchy__c.getValues(SysAdminID);
System.Assert(CS.OverrideMe__c == 'Goodbye');
// Note how this value is null, because you are returning
// data specific for the profile
System.assert(CS.DontOverrideMe__c == null);
```

**Country and State Code Custom Settings Example**

This example illustrates using two custom setting objects for storing related information, and a Visualforce page to display the data in a set of related picklists.

In the following example, country and state codes are stored in two different custom settings: Foundation_Countries and Foundation_States.

The Foundation_Countries custom setting is a list type custom setting and has a single field, Country_Code.
The Foundation_States custom setting is also a List type of custom setting and has the following fields:

- Country Code
- State Code
- State Name

The Visualforce page shows two picklists: one for country and one for state.
The Apex controller `CountryStatePicker` finds the values entered into the custom settings, then returns them to the Visualforce page.

```apex
public with sharing class CountryStatePicker {
    // Variables to store country and state selected by user
```
public String state { get; set; }
public String country {get; set;}

// Generates country dropdown from country settings
public List<SelectOption> getCountriesSelectList() {
    List<SelectOption> options = new List<SelectOption>();
    options.add(new SelectOption('', '-- Select One --'));

    // Find all the countries in the custom setting
    Map<String, Foundation_Countries__c> countries = Foundation_Countries__c.getAll();

    // Sort them by name
    List<String> countryNames = new List<String>();
    countryNames.addAll(countries.keySet());
    countryNames.sort();

    // Create the Select Options.
    for (String countryName : countryNames) {
        Foundation_Countries__c country = countries.get(countryName);
        options.add(new SelectOption(country.country_code__c, country.Name));
    }

    return options;
}

// To generate the states picklist based on the country selected by user.
public List<SelectOption> getStatesSelectList() {
    List<SelectOption> options = new List<SelectOption>();

    // Find all the states we have in custom settings.
    Map<String, Foundation_States__c> allstates = Foundation_States__c.getAll();

    // Filter states that belong to the selected country
    Map<String, Foundation_States__c> states = new Map<String, Foundation_States__c>();
    for (Foundation_States__c state : allstates.values()) {
        if (state.country_code__c == this.country) {
            states.put(state.name, state);
        }
    }

    // Sort the states based on their names
    List<String> stateNames = new List<String>();
    stateNames.addAll(states.keySet());
    stateNames.sort();

    // Generate the Select Options based on the final sorted list
    for (String stateName : stateNames) {
        Foundation_States__c state = states.get(stateName);
        options.add(new SelectOption(state.state_code__c, state.state_name__c));
    }

    // If no states are found, just say not required in the dropdown.
    if (options.size() > 0) {
        options.add(0, new SelectOption('', '-- Select One --'));
    }
IN THIS SECTION:

List Custom Setting Methods
Hierarchy Custom Setting Methods

SEE ALSO:

Apex Developer Guide: Custom Settings

List Custom Setting Methods
The following are instance methods for list custom settings.

IN THIS SECTION:

getAll()
Returns a map of the data sets defined for the custom setting.
getInstance(dataSetName)
Returns the custom setting data set record for the specified data set name. This method returns the exact same object as
getValues(dataSetName).
getValues(dataSetName)
Returns the custom setting data set record for the specified data set name. This method returns the exact same object as
getInstance(dataSetName).

getAll()
Returns a map of the data sets defined for the custom setting.

Signature
public Map<String, CustomSetting__c> getAll()

Return Value
Type: Map<String, CustomSetting__c>

Usage
If no data set is defined, this method returns an empty map.

Note: For Apex saved using Salesforce API version 20.0 or earlier, the data set names, which are the keys in the returned map, are
converted to lower case. For Apex saved using Salesforce API version 21.0 and later, the case of the data set names in the returned
map keys is not changed and the original case is preserved.
getInstance (dataSetName)
Returns the custom setting data set record for the specified data set name. This method returns the exact same object as getValues (dataSetName).

Signature
public CustomSetting__c getInstance(String dataSetName)

Parameters

dataSetName
Type: String

Return Value
Type: CustomSetting__c

Usage
If no data is defined for the specified data set, this method returns null.

getValues (dataSetName)
Returns the custom setting data set record for the specified data set name. This method returns the exact same object as getInstance (dataSetName).

Signature
public CustomSetting__c getValues(String dataSetName)

Parameters

dataSetName
Type: String

Return Value
Type: CustomSetting__c

Usage
If no data is defined for the specified data set, this method returns null.

Hierarchy Custom Setting Methods
The following are instance methods for hierarchy custom settings.
Note:
- In API version 41.0 and below, each method in an Apex test class, including testSetup methods, are able to insert hierarchy custom setting values. This behavior is true even when the methods have the same SetupOwnerId value as a hierarchy custom setting record inserted in a different test method.
- In API version 42.0 and later, if a hierarchy custom setting is inserted in a testSetup method, inserting a hierarchy custom setting record with the same SetupOwnerId in a test method throws a DUPLICATE_VALUE exception.

IN THIS SECTION:

getInstance()
Returns a custom setting data set record for the current user. The fields returned in the custom setting record are merged based on the lowest level fields that are defined in the hierarchy.

getInstance(userId)
Returns the custom setting data set record for the specified user ID. The lowest level custom setting record and fields are returned. Use this when you want to explicitly retrieve data for the custom setting at the user level.

getInstance(profileId)
Returns the custom setting data set record for the specified profile ID. The lowest level custom setting record and fields are returned. Use this when you want to explicitly retrieve data for the custom setting at the profile level.

getOrgDefaults()
Returns the custom setting data set record for the organization.

getValues(userId)
Returns the custom setting data set record for the specified user ID.

getValues(profileId)
Returns the custom setting data set for the specified profile ID.

**getInstance()**

Returns a custom setting data set record for the current user. The fields returned in the custom setting record are merged based on the lowest level fields that are defined in the hierarchy.

**Signature**

```
public CustomSetting__c getInstance()
```

**Return Value**

Type: CustomSetting__c

**Usage**

If no custom setting data is defined for the user, this method returns a new custom setting object. The new custom setting object contains an ID set to `null` and merged fields from higher in the hierarchy. You can add this new custom setting record for the user by using `insert` or `upsert`. If no custom setting data is defined in the hierarchy, the returned custom setting has empty fields, except for the SetupOwnerId field which contains the user ID.
Note: For Apex saved using Salesforce API version 21.0 or earlier, this method returns the custom setting data set record with fields merged from field values defined at the lowest hierarchy level, starting with the user. Also, if no custom setting data is defined in the hierarchy, this method returns `null`.

This method is equivalent to a method call to `getInstance(User_Id)` for the current user.

Example

- Custom setting data set defined for the user: If you have a custom setting data set defined for the user “Uriel Jones,” for the profile “System Administrator,” and for the organization as a whole, and the user running the code is Uriel Jones, this method returns the custom setting record defined for Uriel Jones.

- Merged fields: If you have a custom setting data set with fields A and B for the user “Uriel Jones” and for the profile “System Administrator,” and field A is defined for Uriel Jones, field B is `null` but is defined for the System Administrator profile, this method returns the custom setting record for Uriel Jones with field A for Uriel Jones and field B from the System Administrator profile.

- No custom setting data set record defined for the user: If the current user is “Barbara Mahonie,” who also shares the “System Administrator” profile, but no data is defined for Barbara as a user, this method returns a new custom setting record with the ID set to `null` and with fields merged based on the fields defined in the lowest level in the hierarchy.

`getInstance(userId)`

Returns the custom setting data set record for the specified user ID. The lowest level custom setting record and fields are returned. Use this when you want to explicitly retrieve data for the custom setting at the user level.

**Signature**

```java
public CustomSetting__c getInstance(ID userId)
```

**Parameters**

- `userId`
  
  Type: ID

**Return Value**

Type: CustomSetting__c

**Usage**

If no custom setting data is defined for the user, this method returns a new custom setting object. The new custom setting object contains an ID set to `null` and merged fields from higher in the hierarchy. You can add this new custom setting record for the user by using `insert` or `upsert`. If no custom setting data is defined in the hierarchy, the returned custom setting has empty fields, except for the `SetupOwnerId` field which contains the user ID.

**Note:** For Apex saved using Salesforce API version 21.0 or earlier, this method returns the custom setting data set record with fields merged from field values defined at the lowest hierarchy level, starting with the user. Also, if no custom setting data is defined in the hierarchy, this method returns `null`.

`getInstance(profileId)`

Returns the custom setting data set record for the specified profile ID. The lowest level custom setting record and fields are returned. Use this when you want to explicitly retrieve data for the custom setting at the profile level.
Signature
public CustomSetting__c getInstance(ID profileId)

Parameters
profileId
Type: ID

Return Value
Type: CustomSetting__c

Usage
If no custom setting data is defined for the profile, this method returns a new custom setting record. The new custom setting object contains an ID set to null and with merged fields from your organization's default values. You can add this new custom setting for the profile by using insert or upsert. If no custom setting data is defined in the hierarchy, the returned custom setting has empty fields, except for the SetupOwnerId field which contains the profile ID.

Note: For Apex saved using SalesforceAPI version 21.0 or earlier, this method returns the custom setting data set record with fields merged from field values defined at the lowest hierarchy level, starting with the profile. Also, if no custom setting data is defined in the hierarchy, this method returns null.

getOrgDefaults ()
Returns the custom setting data set record for the organization.

Signature
public CustomSetting__c getOrgDefaults()

Return Value
Type: CustomSetting__c

Usage
If no custom setting data is defined for the organization, this method returns an empty custom setting object.

Note: For Apex saved using Salesforce API version 21.0 or earlier, this method returns null if no custom setting data is defined for the organization.

getValues (userId)
Returns the custom setting data set record for the specified user ID.

Signature
public CustomSetting__c getValues(ID userId)
Parameters

userId
Type: ID

Return Value
Type: CustomSetting__c

Usage
Use this if you only want the subset of custom setting data that has been defined at the user level. For example, suppose you have a custom setting field that has been assigned a value of "alpha" at the organizational level, but has no value assigned at the user or profile level. Using getValues (userId) returns null for this custom setting field.

getValues (profileId)
Returns the custom setting data set for the specified profile ID.

Signature
public CustomSetting__c getValues (ID profileId)

Parameters
profileId
Type: ID

Return Value
Type: CustomSetting__c

Usage
Use this if you only want the subset of custom setting data that has been defined at the profile level. For example, suppose you have a custom setting field that has been assigned a value of "alpha" at the organizational level, but has no value assigned at the user or profile level. Using getValues (ProfileId) returns null for this custom setting field.

Database Class
Contains methods for creating and manipulating data.

Namespace
System
Usage

Some Database methods also exist as DML statements.

SEE ALSO:

Apex DML Operations

Database Methods

The following are methods for Database. All methods are static.

IN THIS SECTION:

convertLead(leadToConvert, allOrNone)
Converts a lead into an account and contact, as well as (optionally) an opportunity.

convertLead(leadsToConvert, allOrNone)
Converts a list of LeadConvert objects into accounts and contacts, as well as (optionally) opportunities.

convertLead(leadToConvert, dmlOptions)
Converts a lead into an account and contact, as well as (optionally) an opportunity.

convertLead(leadsToConvert, dmlOptions)
Converts a list of LeadConvert objects into accounts and contacts, as well as (optionally) opportunities.

convertLead(leadToConvert, allOrNone, accessLevel)
Converts a lead into an account and contact, as well as (optionally) an opportunity.

convertLead(leadsToConvert, allOrNone, accessLevel)
Converts a list of LeadConvert objects into accounts and contacts, as well as (optionally) opportunities.

convertLead(leadToConvert, dmlOptions, accessLevel)
Converts a lead into an account and contact, as well as (optionally) an opportunity.

convertLead(leadsToConvert, dmlOptions, accessLevel)
Converts a list of LeadConvert objects into accounts and contacts, as well as (optionally) opportunities.

countQuery(query)
Returns the number of records that a dynamic SOQL query would return when executed.

countQuery(query, accessLevel)
Returns the number of records that a dynamic SOQL query would return when executed.

countQueryWithBinds(query, bindMap, accessLevel)
Returns the number of records that a dynamic SOQL query would return when executed. Bind variables in the query are resolved from the bindMap Map parameter directly with the key, rather than from Apex code variables.

delete(recordToDelete, allOrNone)
Deletes an existing sObject record, such as an individual account or contact, from your organization's data.

delete(recordsToDelete, allOrNone)
Deletes a list of existing sObject records, such as individual accounts or contacts, from your organization's data.

delete(recordID, allOrNone)
Deletes existing sObject records, such as individual accounts or contacts, from your organization's data.
delete(recordIDs, allOrNone)
Deletes a list of existing sObject records, such as individual accounts or contacts, from your organization’s data.

delete(recordToDelete, allOrNone, accessLevel)
Deletes an existing sObject record, such as an individual account or contact, from your organization’s data.

delete(recordsToDelete, allOrNone, accessLevel)
Deletes a list of existing sObject records, such as individual accounts or contacts, from your organization’s data.

delete(recordID, allOrNone, accessLevel)
Deletes existing sObject records, such as individual accounts or contacts, from your organization’s data.

delete(recordIDs, allOrNone, accessLevel)
Deletes a list of existing sObject records, such as individual accounts or contacts, from your organization’s data.

deleteAsync(sobjects, callback)
Initiates requests to delete the external data that corresponds to the specified external object records. The request is executed asynchronously, as a background operation, and is sent to the external system that’s defined by the external object’s associated external data source. Allows referencing a callback class whose processDelete method is called for each record after deletion.

deleteAsync(sobject, callback)
Initiates a request to delete the external data that corresponds to the specified external object record. The request is executed asynchronously, as a background operation, and is sent to the external system that’s defined by the external object’s associated external data source. Allows referencing a callback class whose processDelete method is called after deletion.

deleteAsync(sobjects)
Initiates requests to delete the external data that corresponds to the specified external object records. The requests are executed asynchronously, as background operations, and are sent to the external systems that are defined by the external objects’ associated external data sources.

deleteAsync(sobject)
Initiates a request to delete the external data that corresponds to the specified external object record. The request is executed asynchronously, as a background operation, and is sent to the external system that’s defined by the external object’s associated external data source.

deleteAsync(sobjects, callback, accessLevel)
Initiates requests to delete the external data that corresponds to the specified external object records. The request is executed asynchronously, as a background operation, and is sent to the external system that’s defined by the external object’s associated external data source. Allows referencing a callback class whose processDelete method is called for each record after deletion.

deleteAsync(sobject, callback, accessLevel)
Initiates a request to delete the external data that corresponds to the specified external object record. The request is executed asynchronously, as a background operation, and is sent to the external system that’s defined by the external object’s associated external data source. Allows referencing a callback class whose processDelete method is called after deletion.

deleteAsync(sobjects, accessLevel)
Initiates requests to delete the external data that corresponds to the specified external object records. The requests are executed asynchronously, as background operations, and are sent to the external systems that are defined by the external objects’ associated external data sources.

deleteAsync(sobject, accessLevel)
Initiates a request to delete the external data that corresponds to the specified external object record. The request is executed asynchronously, as a background operation, and is sent to the external system that’s defined by the external object’s associated external data source.
deleteImmediate(sobjects)
Initiates requests to delete the external data that corresponds to the specified external object records. The requests are executed synchronously and are sent to the external systems that are defined by the external objects' associated external data sources. If the Apex transaction contains pending changes, the synchronous operations can’t be completed and throw exceptions.

deleteImmediate(sobject)
Initiates a request to delete the external data that corresponds to the specified external object record. The request is executed synchronously and is sent to the external system that's defined by the external object’s associated external data source. If the Apex transaction contains pending changes, the synchronous operation can’t be completed and throws an exception.

deleteImmediate(sobjects, accessLevel)
Initiates requests to delete the external data that corresponds to the specified external object records. The requests are executed synchronously and are sent to the external systems that are defined by the external objects' associated external data sources. If the Apex transaction contains pending changes, the synchronous operations can’t be completed and throw exceptions.

deleteImmediate(sobject, accessLevel)
Initiates a request to delete the external data that corresponds to the specified external object record. The request is executed synchronously and is sent to the external system that’s defined by the external object’s associated external data source. If the Apex transaction contains pending changes, the synchronous operation can’t be completed and throws an exception.

emptyRecycleBin(recordIds)
Permanently deletes the specified records from the Recycle Bin.

emptyRecycleBin(sObject)
Permanently deletes the specified sObject from the Recycle Bin.

emptyRecycleBin(listOfSObjects)
Permanently deletes the specified sObjects from the Recycle Bin.

executeBatch(batchClassObject)
Submits a batch Apex job for execution corresponding to the specified class.

executeBatch(batchClassObject, scope)
Submits a batch Apex job for execution using the specified class and scope.

getAsyncDeleteResult(deleteResult)
Retrieves the status of an asynchronous delete operation that’s identified by a Database.DeleteResult object.

getAsyncDeleteResult(asyncLocator)
Retrieves the result of an asynchronous delete operation based on the result’s unique identifier.

getAsyncLocator(result)
Returns the asyncLocator associated with the result of a specified asynchronous insert, update, or delete operation.

getAsyncSaveResult(saveResult)
Returns the status of an asynchronous insert or update operation that’s identified by a Database.SaveResult object.

getAsyncSaveResult(asyncLocator)
Returns the status of an asynchronous insert or update operation based on the unique identifier associated with each modification.

getDeleted(sObjectType, startDate, endDate)
Returns the list of individual records that have been deleted for an sObject type within the specified start and end dates and times and that are still in the Recycle Bin.

getQueryLocator(listOfQueries)
Creates a QueryLocator object used in batch Apex or Visualforce.
**getQueryLocator(query)**
Creates a QueryLocator object used in batch Apex or Visualforce.

**getQueryLocator(listOfQueries, accessLevel)**
Creates a QueryLocator object used in batch Apex or Visualforce.

**getQueryLocator(query, accessLevel)**
Creates a QueryLocator object used in batch Apex or Visualforce.

**getQueryLocatorWithBinds(query, bindMap, accessLevel)**
Creates a QueryLocator object used in batch Apex or Visualforce. Bind variables in the query are resolved from the bindMap Map parameter directly with the key, rather than from Apex code variables.

**getUpdated(sobjectType, startDate, endDate)**
Returns the list of individual records that have been updated for an sObject type within the specified start and end dates and times.

**insert(recordToInsert, allOrNone)**
Adds an sObject, such as an individual account or contact, to your organization's data.

**insert(recordsToInsert, allOrNone)**
Adds one or more sObjects, such as individual accounts or contacts, to your organization's data.

**insert(recordToInsert, dmlOptions)**
Adds an sObject, such as an individual account or contact, to your organization's data.

**insert(recordsToInsert, dmlOptions)**
Adds one or more sObjects, such as individual accounts or contacts, to your organization's data.

**insert(recordToInsert, allOrNone, accessLevel)**
Adds an sObject, such as an individual account or contact, to your organization's data.

**insert(recordsToInsert, allOrNone, accessLevel)**
Adds one or more sObjects, such as individual accounts or contacts, to your organization's data.

**insert(recordToInsert, dmlOptions, accessLevel)**
Adds an sObject, such as an individual account or contact, to your organization's data.

**insert(recordsToInsert, dmlOptions, accessLevel)**
Adds one or more sObjects, such as individual accounts or contacts, to your organization's data.

**insertAsync(sobjects, callback)**
Initiates requests to add external object data to the relevant external systems. The requests are executed asynchronously, as background operations, and are sent to the external systems that are defined by the external objects' associated external data sources. Allows referencing a callback class whose processSave method is called for each record after the remote operations are completed.

**insertAsync(sobject, callback)**
Initiates a request to add external object data to the relevant external system. The request is executed asynchronously, as a background operation, and is sent to the external system that's defined by the external object's associated external data source. Allows referencing a callback class whose processSave method is called after the remote operation is completed.

**insertAsync(sobjects)**
Initiates requests to add external object data to the relevant external systems. The requests are executed asynchronously, as background operations, and are sent to the external systems that are defined by the external objects' associated external data sources.

**insertAsync(sobject)**
Initiates a request to add external object data to the relevant external system. The request is executed asynchronously, as a background operation, and is sent to the external system that's defined by the external object's associated external data source.
insertAsync(sobjects, callback, accessLevel)
Initiates requests to add external object data to the relevant external systems. The requests are executed asynchronously, as background operations, and are sent to the external systems that are defined by the external objects' associated external data sources. Allows referencing a callback class whose processSave method is called for each record after the remote operations are completed.

insertAsync(object, callback, accessLevel)
Initiates a request to add external object data to the relevant external system. The request is executed asynchronously, as a background operation, and is sent to the external system that's defined by the external object's associated external data source. Allows referencing a callback class whose processSave method is called after the remote operation is completed.

insertAsync(sobjects, accessLevel)
Initiates requests to add external object data to the relevant external systems. The requests are executed asynchronously, as background operations, and are sent to the external systems that are defined by the external objects' associated external data sources.

insertAsync(object, accessLevel)
Initiates a request to add external object data to the relevant external system. The request is executed asynchronously, as a background operation, and is sent to the external system that's defined by the external object's associated external data source.

insertImmediate(sobjects)
Initiates requests to add external object data to the relevant external systems. The requests are executed synchronously and are sent to the external systems that are defined by the external objects' associated external data sources. If the Apex transaction contains pending changes, the synchronous operations can't be completed and throw exceptions.

insertImmediate(object)
Initiates a request to add external object data to the relevant external system. The request is executed synchronously and is sent to the external system that's defined by the external object's associated external data source. If the Apex transaction contains pending changes, the synchronous operation can't be completed and throws an exception.

insertImmediate(sobjects, accessLevel)
Initiates requests to add external object data to the relevant external systems. The requests are executed synchronously and are sent to the external systems that are defined by the external objects' associated external data sources. If the Apex transaction contains pending changes, the synchronous operations can't be completed and throw exceptions.

insertImmediate(object, accessLevel)
Initiates a request to add external object data to the relevant external system. The request is executed synchronously and is sent to the external system that's defined by the external object's associated external data source. If the Apex transaction contains pending changes, the synchronous operation can't be completed and throws an exception.

merge(masterRecord, duplicateId)
Merges the specified duplicate record into the master sObject record of the same type, deleting the duplicate, and reparenting any related records. Merges only accounts, contacts, or leads.

merge(masterRecord, duplicateRecord)
Merges the specified duplicate sObject record into the master sObject of the same type, deleting the duplicate, and reparenting any related records.

merge(masterRecord, duplicateIds)
Merges up to two records of the same sObject type into the master sObject record, deleting the others, and reparenting any related records.

merge(masterRecord, duplicateRecords)
Merges up to two records of the same object type into the master sObject record, deleting the others, and reparenting any related records.
merge(masterRecord, duplicateId, allOrNone)
Merges the specified duplicate record into the master sObject record of the same type, optionally returning errors, if any, deleting the duplicate, and reparenting any related records. Merges only accounts, contacts, or leads.

merge(masterRecord, duplicateRecord, allOrNone)
Merges the specified duplicate sObject record into the master sObject of the same type, optionally returning errors, if any, deleting the duplicate, and reparenting any related records.

merge(masterRecord, duplicateIds, allOrNone)
Merges up to two records of the same sObject type into the master sObject record, optionally returning errors, if any, deleting the duplicates, and reparenting any related records.

merge(masterRecord, duplicateRecords, allOrNone)
Merges up to two records of the same object type into the master sObject record, optionally returning errors, if any, deleting the duplicates, and reparenting any related records.

merge(masterRecord, duplicateId, accessLevel)
Merges the specified duplicate record into the master sObject record of the same type, deleting the duplicate, and reparenting any related records. Merges only accounts, contacts, or leads.

merge(masterRecord, duplicateRecord, accessLevel)
Merges the specified duplicate sObject record into the master sObject of the same type, deleting the duplicate, and reparenting any related records.

merge(masterRecord, duplicateIds, accessLevel)
Merges up to two records of the same sObject type into the master sObject record, deleting the others, and reparenting any related records.

merge(masterRecord, duplicateRecords, accessLevel)
Merges up to two records of the same object type into the master sObject record, deleting the others, and reparenting any related records.

merge(masterRecord, duplicateId, allOrNone, accessLevel)
Merges the specified duplicate record into the master sObject record of the same type, optionally returning errors, if any, deleting the duplicate, and reparenting any related records. Merges only accounts, contacts, or leads.

merge(masterRecord, duplicateRecord, allOrNone, accessLevel)
Merges the specified duplicate sObject record into the master sObject of the same type, optionally returning errors, if any, deleting the duplicate, and reparenting any related records.

merge(masterRecord, duplicateIds, allOrNone, accessLevel)
Merges up to two records of the same sObject type into the master sObject record, optionally returning errors, if any, deleting the duplicates, and reparenting any related records.

merge(masterRecord, duplicateRecords, allOrNone, accessLevel)
Merges up to two records of the same object type into the master sObject record, optionally returning errors, if any, deleting the duplicates, and reparenting any related records.

query(queryString)
Creates a dynamic SOQL query at runtime.

query(queryString, accessLevel)
Creates a dynamic SOQL query at runtime.

queryWithBinds(queryString, bindMap, accessLevel)
Creates a dynamic SOQL query at runtime. Bind variables in the query are resolved from the $bindMap Map parameter directly with the key, rather than from Apex code variables.
**rollback(databaseSavepoint)**
Restores the database to the state specified by the savepoint variable. Any emails submitted since the last savepoint are also rolled back and not sent.

**setSavepoint()**
Returns a savepoint variable that can be stored as a local variable, then used with the `rollback` method to restore the database to that point.

**undelete(recordToUndelete, allOrNone)**
Restores an existing sObject record, such as an individual account or contact, from your organization's Recycle Bin.

**undelete(recordsToUndelete, allOrNone)**
Restores one or more existing sObject records, such as individual accounts or contacts, from your organization's Recycle Bin.

**undelete(recordID, allOrNone)**
Restores an existing sObject record, such as an individual account or contact, from your organization's Recycle Bin.

**undelete(recordIDs, allOrNone)**
Restores one or more existing sObject records, such as individual accounts or contacts, from your organization's Recycle Bin.

**undelete(recordToUndelete, allOrNone, accessLevel)**
Restores an existing sObject record, such as an individual account or contact, from your organization's Recycle Bin.

**undelete(recordsToUndelete, allOrNone, accessLevel)**
Restores one or more existing sObject records, such as individual accounts or contacts, from your organization's Recycle Bin.

**undelete(recordID, allOrNone, accessLevel)**
Restores an existing sObject record, such as an individual account or contact, from your organization's Recycle Bin.

**undelete(recordIDs, allOrNone, accessLevel)**
Restores one or more existing sObject records, such as individual accounts or contacts, from your organization's Recycle Bin.

**update(recordToUpdate, allOrNone)**
Modifies an existing sObject record, such as an individual account or contact, in your organization's data.

**update(recordsToUpdate, allOrNone)**
Modifies one or more existing sObject records, such as individual accounts or contacts, in your organization's data.

**update(recordToUpdate, dmlOptions)**
Modifies an existing sObject record, such as an individual account or contact, in your organization's data.

**update(recordsToUpdate, dmlOptions)**
Modifies one or more existing sObject records, such as individual accounts or contacts, in your organization's data.

**update(recordToUpdate, allOrNone, accessLevel)**
Modifies an existing sObject record, such as an individual account or contact, in your organization's data.

**update(recordsToUpdate, allOrNone, accessLevel)**
Modifies one or more existing sObject records, such as individual accounts or contacts, in your organization's data.

**update(recordToUpdate, dmlOptions, accessLevel)**
Modifies an existing sObject record, such as an individual account or contact, in your organization's data.

**update(recordsToUpdate, dmlOptions, accessLevel)**
Modifies one or more existing sObject records, such as individual accounts or contacts, in your organization's data.

**upsert(recordToUpsert, externalIdField, allOrNone)**
Creates a new sObject record or updates an existing sObject record within a single statement, using a specified field to determine the presence of existing objects, or the ID field if no field is specified.
upsert(recordsToUpsert, externalIdField, allOrNone)
Creates new sObject records or updates existing sObject records within a single statement, using a specified field to determine the presence of existing objects, or the ID field if no field is specified.

upsert(recordToUpsert, externalIdField, allOrNone, accessLevel)
Creates a new sObject record or updates an existing sObject record within a single statement, using a specified field to determine the presence of existing objects, or the ID field if no field is specified.

upsert(recordsToUpsert, externalIdField, allOrNone, accessLevel)
Creates new sObject records or updates existing sObject records within a single statement, using a specified field to determine the presence of existing objects, or the ID field if no field is specified.

updateAsync(sobjects, callback)
Initiates requests to update external object data on the relevant external systems. The requests are executed asynchronously, as background operations, and are sent to the external systems that are defined by the external objects’ associated external data sources. Allows referencing a callback class whose processSave method is called for each record after the remote operations are completed.

updateAsync(sobject, callback)
Initiates a request to update external object data on the relevant external system. The request is executed asynchronously, as a background operation, and is sent to the external system that’s defined by the external object’s associated external data source. Allows referencing a callback class whose processSave method is called after the remote operation is completed.

updateAsync(sobjects)
Initiates requests to update external object data on the relevant external systems. The requests are executed asynchronously, as background operations, and are sent to the external systems that are defined by the external objects’ associated external data sources.

updateAsync(sobject)
Initiates a request to update external object data on the relevant external system. The request is executed asynchronously, as a background operation, and is sent to the external system that’s defined by the external object’s associated external data source.

updateAsync(sobjects, callback, accessLevel)
Initiates requests to update external object data on the relevant external systems. The requests are executed asynchronously, as background operations, and are sent to the external systems that are defined by the external objects’ associated external data sources. Allows referencing a callback class whose processSave method is called for each record after the remote operations are completed.

updateAsync(sobject, callback, accessLevel)
Initiates a request to update external object data on the relevant external system. The request is executed asynchronously, as a background operation, and is sent to the external system that’s defined by the external object’s associated external data source. Allows referencing a callback class whose processSave method is called after the remote operation is completed.

updateAsync(sobjects, accessLevel)
Initiates requests to update external object data on the relevant external systems. The requests are executed asynchronously, as background operations, and are sent to the external systems that are defined by the external objects’ associated external data sources.

updateAsync(sobject, accessLevel)
Initiates a request to update external object data on the relevant external system. The request is executed asynchronously, as a background operation, and is sent to the external system that’s defined by the external object’s associated external data source.

updateImmediate(sobjects)
Initiates requests to update external object data on the relevant external systems. The requests are executed synchronously and are sent to the external systems that are defined by the external objects’ associated external data sources. If the Apex transaction contains pending changes, the synchronous operations can’t be completed and throw exceptions.
updateImmediate(sobject)
Initiates a request to update external object data on the relevant external system. The request is executed synchronously and is sent to the external system that’s defined by the external object’s associated external data source. If the Apex transaction contains pending changes, the synchronous operation can’t be completed and throws an exception.

updateImmediate(sobjects, accessLevel)
Initiates requests to update external object data on the relevant external systems. The requests are executed synchronously and are sent to the external systems that are defined by the external objects’ associated external data sources. If the Apex transaction contains pending changes, the synchronous operations can’t be completed and throw exceptions.

updateImmediate(sobject, accessLevel)
Initiates a request to update external object data on the relevant external system. The request is executed synchronously and is sent to the external system that’s defined by the external object’s associated external data source. If the Apex transaction contains pending changes, the synchronous operation can’t be completed and throws an exception.

convertLead(leadToConvert, allOrNone)
Converts a lead into an account and contact, as well as (optionally) an opportunity.

Signature
public static Database.LeadConvertResult convertLead(Database.LeadConvert leadToConvert, Boolean allOrNone)

Parameters
leadToConvert
Type: Database.LeadConvert

callOrNone
Type: Boolean

The optional allOrNone parameter specifies whether the operation allows partial success. If you specify false for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set true, an exception is thrown if the method is not successful.

Return Value
Type: Database.LeadConvertResult

Usage
The convertLead method accepts up to 100 LeadConvert objects.
Each executed convertLead method counts against the governor limit for DML statements.

convertLead(leadsToConvert, allOrNone)
Converts a list of LeadConvert objects into accounts and contacts, as well as (optionally) opportunities.
Signature

public static Database.LeadConvertResult[] convertLead(Database.LeadConvert[] leadsToConvert, Boolean allOrNone)

Parameters

leadsToConvert
Type: Database.LeadConvert[]

allOrNone
Type: Boolean

The optional allOrNone parameter specifies whether the operation allows partial success. If you specify false for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set true, an exception is thrown if the method is not successful.

Return Value

Type: Database.LeadConvertResult[]

Usage

The convertLead method accepts up to 100 LeadConvert objects. Each executed convertLead method counts against the governor limit for DML statements.

convertLead(leadToConvert, dmlOptions)

Converts a lead into an account and contact, as well as (optionally) an opportunity.

Signature

public static Database.LeadConvertResult convertLead(Database.LeadConvert leadToConvert, Database.DMLOptions dmlOptions)

Parameters

leadToConvert
Type: Database.LeadConvert

dmlOptions
Type: Database.DMLOptions

The optional dmlOptions parameter specifies additional data for the transaction, such as assignment rule information or rollback behavior when errors occur during record insertions.

Return Value

Type: Database.LeadConvertResult
Usage

The `convertLead` method accepts up to 100 `LeadConvert` objects. Each executed `convertLead` method counts against the governor limit for DML statements.

`convertLead(leadToConvert, allOrNone, accessLevel)`

Converts a lead into an account and contact, as well as (optionally) an opportunity.

Signature

```
public static Database.LeadConvertResult convertLead(Database.LeadConvert leadToConvert,
                                                      Boolean allOrNone, System.AccessLevel accessLevel)
```

Parameters

`leadToConvert`
- Type: `Database.LeadConvert`

`allOrNone`
- Type: `Boolean`

The optional `allOrNone` parameter specifies whether the operation allows partial success. If you specify `false` for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to
verify which records succeeded, which failed, and why. If the parameter is not set or is set true, an exception is thrown if the method is not successful.

accessLevel
Type: System.AccessLevel
(Optional) The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value
Type: Database.LeadConvertResult

Usage
The convertLead method accepts up to 100 LeadConvert objects. Each executed convertLead method counts against the governor limit for DML statements.

convertLead(leadstoConvert, allOrNone, accessLevel)
Converts a list of LeadConvert objects into accounts and contacts, as well as (optionally) opportunities.

Signature
public static List<Database.LeadConvertResult> convertLead(List<Database.LeadConvert> leadstoConvert, Boolean allOrNone, System.AccessLevel accessLevel)

Parameters
leadstoConvert
Type: List<Database.LeadConvert>
allOrNone
Type: Boolean
The optional allOrNone parameter specifies whether the operation allows partial success. If you specify false for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set true, an exception is thrown if the method is not successful.

accessLevel
Type: System.AccessLevel
(Optional) The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value
Type: List<Database.LeadConvertResult>
**Usage**

The `convertLead` method accepts up to 100 `LeadConvert` objects. Each executed `convertLead` method counts against the governor limit for DML statements.

\[
\text{convertLead(leadToConvert, dmlOptions, accessLevel)}
\]

Converts a lead into an account and contact, as well as (optionally) an opportunity.

**Signature**

```
public static Database.LeadConvertResult convertLead(Database.LeadConvert leadToConvert, 
                                                  Database.DMLOptions dmlOptions, System.AccessLevel accessLevel)
```

**Parameters**

- **leadToConvert**
  - Type: `Database.LeadConvert`

- **dmlOptions**
  - Type: `Database.DMLOptions`
    - The optional `dmlOptions` parameter specifies additional data for the transaction, such as assignment rule information or rollback behavior when errors occur during record insertions.

- **accessLevel**
  - Type: `System.AccessLevel`
    - (Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

**Return Value**

Type: `Database.LeadConvertResult`

**Usage**

The `convertLead` method accepts up to 100 `LeadConvert` objects. Each executed `convertLead` method counts against the governor limit for DML statements.

\[
\text{convertLead(leadsToConvert, dmlOptions, accessLevel)}
\]

Converts a list of `LeadConvert` objects into accounts and contacts, as well as (optionally) opportunities.

**Signature**

```
public static List<Database.LeadConvertResult> convertLead(List<Database.LeadConvert> 
                            leadsToConvert, Database.DMLOptions dmlOptions, System.AccessLevel accessLevel)
```
Parameters

`leadsToConvert`
Type: List<Database.LeadConvert>

`dmlOptions`
Type: Database.DMLOptions

The optional `dmlOptions` parameter specifies additional data for the transaction, such as assignment rule information or rollback behavior when errors occur during record insertions.

`accessLevel`
Type: System.AccessLevel

(Optional) The `accessLevel` parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value
Type: List<Database.LeadConvertResult>

Usage

The `convertLead` method accepts up to 100 `LeadConvert` objects.

Each executed `convertLead` method counts against the governor limit for DML statements.

`countQuery(query)`

Returns the number of records that a dynamic SOQL query would return when executed.

Signature

`public static Integer countQuery(String query)`

Parameters

`query`
Type: String

Return Value
Type: Integer

Usage

For more information, see Dynamic SOQL.

Each executed `countQuery` method counts against the governor limit for SOQL queries.
Example

```java
String queryString = 'SELECT count() FROM Account';
Integer i = Database.countQuery(queryString);
```

countQuery(query, accessLevel)

Returns the number of records that a dynamic SOQL query would return when executed.

Signature

```java
public static Integer countQuery(String query, System.AccessLevel accessLevel)
```

Parameters

- `query`
  - Type: `String`
- `accessLevel`
  - Type: `System.AccessLevel`

  (Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value

- Type: `Integer`

Usage

For more information, see Dynamic SOQL.

Each executed `countQuery` method counts against the governor limit for SOQL queries.

countQueryWithBinds(query, bindMap, accessLevel)

Returns the number of records that a dynamic SOQL query would return when executed. Bind variables in the query are resolved from the `bindMap` Map parameter directly with the key, rather than from Apex code variables.

Signature

```java
public static Integer countQueryWithBinds(String query, Map<String, Object> bindMap, System.AccessLevel accessLevel)
```

Parameters

- `query`
  - Type: `String`
SOQL query that includes Apex bind variables preceded by a colon. All bind variables must have a key in the `bindMap` Map.

`bindMap`
Type: `Map<String, Object>`
A map that contains keys for each bind variable specified in the SOQL `queryString` and its value. The keys can’t be null or duplicates, and the values can’t be null or empty strings.

`accessLevel`
Type: `System.AccessLevel`
The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced.

**Return Value**
Type: `Integer`

**Usage**
For more information, see Dynamic SOQL.
Each executed `countQueryWithBinds` method counts against the governor limit for SOQL queries.

**Example**
In this example, the SOQL query uses a bind variable for an Account name. Its value (`Acme Inc.`) is passed in to the method using the `nameBind` Map. The `accountName` variable isn’t (and doesn’t have to be) in scope when the query is executed within the method.

```java
public static Integer simpleBindingSoqlQuery(Map<String, Object> bindParams) {
    String queryString =
        'SELECT count() ' +
        'FROM Account ' +
        'WHERE name = :name';
    return Database.countQueryWithBinds(
        queryString,
        bindParams,
        AccessLevel.USER_MODE
    );
}

String accountName = 'Acme Inc.';
Map<String, Object> nameBind = new Map<String, Object>('name' => accountName);
Integer acctCount = simpleBindingSoqlQuery(nameBind);
System.debug(acctCount);
```

`delete(recordToDelete, allOrNone)`
Deletes an existing sObject record, such as an individual account or contact, from your organization’s data.
public static Database.DeleteResult delete(SObject recordToDelete, Boolean allOrNone)

Parameters

recordToDelete
Type: sObject

allOrNone
Type: Boolean

The optional allOrNone parameter specifies whether the operation allows partial success. If you specify false for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set true, an exception is thrown if the method is not successful.

Return Value

Type: Database.DeleteResult

Usage

delete is analogous to the delete() statement in the SOAP API.

Each executed delete method counts against the governor limit for DML statements.

delete(recordsToDelete, allOrNone)

Deletes a list of existing sObject records, such as individual accounts or contacts, from your organization’s data.

Signature

public static Database.DeleteResult[] delete(SObject[] recordsToDelete, Boolean allOrNone)

Parameters

recordsToDelete
Type: sObject[]

allOrNone
Type: Boolean

The optional allOrNone parameter specifies whether the operation allows partial success. If you specify false for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set true, an exception is thrown if the method is not successful.

Return Value

Type: Database.DeleteResult[]
Usage

delete is analogous to the delete() statement in the SOAP API.
Each executed delete method counts against the governor limit for DML statements.

Example

The following example deletes an account named 'DotCom':

```java
Account[] doomedAccts = [SELECT Id, Name FROM Account WHERE Name = 'DotCom'];
Database.DeleteResult[] DR_Dels = Database.delete(doomedAccts);
```

delete(recordID, allOrNone)

Deletes existing sObject records, such as individual accounts or contacts, from your organization's data.

Signature

```java
public static Database.DeleteResult delete(ID recordID, Boolean allOrNone)
```

Parameters

- **recordID**
  - Type: ID
- **allOrNone**
  - Type: Boolean

The optional allOrNone parameter specifies whether the operation allows partial success. If you specify false for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set true, an exception is thrown if the method is not successful.

Return Value

Type: Database.DeleteResult

Usage

delete is analogous to the delete() statement in the SOAP API.
Each executed delete method counts against the governor limit for DML statements.

delete(recordIDs, allOrNone)

Deletes a list of existing sObject records, such as individual accounts or contacts, from your organization's data.

Signature

```java
public static Database.DeleteResult[] delete(ID[] recordIDs, Boolean allOrNone)
```
Parameters

recordIDs
  Type: ID[]

allOrNone
  Type: Boolean

  The optional allOrNone parameter specifies whether the operation allows partial success. If you specify false for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set true, an exception is thrown if the method is not successful.

Return Value

Type: Database.DeleteResult[]

Usage

delete is analogous to the delete() statement in the SOAP API.

Each executed delete method counts against the governor limit for DML statements.

delete(recordToDelete, allOrNone, accessLevel)

Deletes an existing sObject record, such as an individual account or contact, from your organization’s data.

Signature

public static Database.DeleteResult delete(SObject recordToDelete, Boolean allOrNone, System.AccessLevel accessLevel)

Parameters

recordToDelete
  Type: SObject

allOrNone
  Type: Boolean

  The optional allOrNone parameter specifies whether the operation allows partial success. If you specify false for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set true, an exception is thrown if the method is not successful.

accessLevel
  Type: System.AccessLevel

  (Optional) The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.
Return Value
Type: Database.DeleteResult

Usage
delete is analogous to the delete() statement in the SOAP API.
Each executed delete method counts against the governor limit for DML statements.

delete(recordsToDelete, allOrNone, accessLevel)
Deletes a list of existing sObject records, such as individual accounts or contacts, from your organization’s data.

Signature
public static List<Database.DeleteResult> delete(List<SObject> recordsToDelete, Boolean allOrNone, System.AccessLevel accessLevel)

Parameters
recordsToDelete
Type: List<sObject>

allOrNone
Type: Boolean
The optional allOrNone parameter specifies whether the operation allows partial success. If you specify false for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set true, an exception is thrown if the method is not successful.

accessLevel
Type: System.AccessLevel
(Optional) The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value
Type: List<Database.DeleteResult>

Usage
delete is analogous to the delete() statement in the SOAP API.
Each executed delete method counts against the governor limit for DML statements.

delete(recordID, allOrNone, accessLevel)
Deletes existing sObject records, such as individual accounts or contacts, from your organization’s data.
public static Database.DeleteResult delete(Id recordID, Boolean allOrNone, System.AccessLevel accessLevel)

Parameters

recordID
Type: ID

allOrNone
Type: Boolean

The optional allOrNone parameter specifies whether the operation allows partial success. If you specify false for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set true, an exception is thrown if the method is not successful.

accessLevel
Type: System.AccessLevel

(Optional) The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value

Type: Database.DeleteResult

Usage

delete is analogous to the delete() statement in the SOAP API.

Each executed delete method counts against the governor limit for DML statements.

delete(recordIDs, allOrNone, accessLevel)

Deletes a list of existing sObject records, such as individual accounts or contacts, from your organization’s data.

public static List<Database.DeleteResult> delete(List<Id> recordIDs, Boolean allOrNone, System.AccessLevel accessLevel)

Parameters

recordIDs
Type: List<Id>

allOrNone
Type: Boolean

The optional allOrNone parameter specifies whether the operation allows partial success. If you specify false for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to
verify which records succeeded, which failed, and why. If the parameter is not set or is set true, an exception is thrown if the method is not successful.

accessLevel
Type: System.AccessLevel

(Optional) The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value
Type: List<Database.DeleteResult>

Usage
delete is analogous to the delete() statement in the SOAP API.
Each executed delete method counts against the governor limit for DML statements.

deleteAsync(sobjects, callback)
_Initiates requests to delete the external data that corresponds to the specified external object records. The request is executed asynchronously, as a background operation, and is sent to the external system that’s defined by the external object’s associated external data source. Allows referencing a callback class whose processDelete method is called for each record after deletion._

Signature

public static List<Database.DeleteResult> deleteAsync(List<SObject> sobjects, DataSource.AsyncDeleteCallback callback)

Parameters

sobjects
Type: List<SObject>
List of external object records to delete.

callback
Type: DataSource.AsyncDeleteCallback
The callback that contains the state in the originating context and an action (the processDelete method) that is executed after the insert operation is completed. Use the action callback to update org data according to the operation’s results. The callback object must extend DataSource.AsyncDeleteCallback.

Return Value
Type: List<Database.DeleteResult>
Status results for the delete operation. Each result corresponds to a record processed by this asynchronous operation and is associated with a unique identifier (asyncLocator). The asyncLocator value is included in the errors array of the result. You can retrieve this identifier with Database.getAsyncLocator(). Retrieve the final result with Database.getAsyncDeleteResult().
**deleteAsync(sobject, callback)**

Initiates a request to delete the external data that corresponds to the specified external object record. The request is executed asynchronously, as a background operation, and is sent to the external system that’s defined by the external object’s associated external data source. Allows referencing a callback class whose `processDelete` method is called after deletion.

**Signature**

```java
public static Database.DeleteResult deleteAsync(SObject sobject, DataSource.AsyncDeleteCallback callback)
```

**Parameters**

- **sobject**
  - Type: `SObject`
  - The external object record to delete.

- **callback**
  - Type: `DataSource.AsyncDeleteCallback`
  - The callback that contains the state in the originating context and an action (the `processDelete` method) that is executed after the insert operation is completed. Use the action callback to update org data according to the operation’s results. The callback object must extend `DataSource.AsyncDeleteCallback`.

**Return Value**

Type: `Database.DeleteResult`

Status result for the delete operation. The result corresponds to the record processed by this asynchronous operation and is associated with a unique identifier (`asyncLocator`). The `asyncLocator` value is included in the errors array of the result. You can retrieve this identifier with `Database.getAsyncLocator()`. Retrieve the final result with `Database.getAsyncDeleteResult()`.

**deleteAsync(sobjects)**

Initiates requests to delete the external data that corresponds to the specified external object records. The requests are executed asynchronously, as background operations, and are sent to the external systems that are defined by the external objects’ associated external data sources.

**Signature**

```java
public static List<Database.DeleteResult> deleteAsync(List<SObject> sobjects)
```

**Parameters**

- **sobjects**
  - Type: `List<SObject>`
  - List of external object records to delete.

**Return Value**

Type: `List<Database.DeleteResult>`
Status results for the delete operation. Each result corresponds to a record processed by this asynchronous operation and is associated with a unique identifier (asyncLocator). The asyncLocator value is included in the errors array of the result. You can retrieve this identifier with `Database.getAsyncLocator()`. Retrieve the final result with `Database.getAsyncDeleteResult()`.

**deleteAsync(sobject)**

Initiates a request to delete the external data that corresponds to the specified external object record. The request is executed asynchronously, as a background operation, and is sent to the external system that's defined by the external object's associated external data source.

**Signature**

```java
public static Database.DeleteResult deleteAsync(SObject sobject)
```

**Parameters**

`sobject`

Type: `SObject`

The external object record to delete.

**Return Value**

Type: `Database.DeleteResult`

Status result for the delete operation. The result corresponds to the record processed by this asynchronous operation and is associated with a unique identifier (asyncLocator). The asyncLocator value is included in the errors array of the result. You can retrieve this identifier with `Database.getAsyncLocator()`. Retrieve the final result with `Database.getAsyncDeleteResult()`.

**deleteAsync(sobjects, callback, accessLevel)**

Initiates requests to delete the external data that corresponds to the specified external object records. The request is executed asynchronously, as a background operation, and is sent to the external system that's defined by the external object's associated external data source. Allows referencing a callback class whose `processDelete` method is called for each record after deletion.

**Signature**

```java
public static List<Database.DeleteResult> deleteAsync(List<SObject> sobjects,
Data Source.AsyncDeleteCallback callback, System.AccessLevel accessLevel)
```

**Parameters**

`sobjects`

Type: `List<SObject>`

List of external object records to delete.

`callback`

Type: `DataSource.AsyncDeleteCallback`

The callback that contains the state in the originating context and an action (the `processDelete` method) that is executed after the insert operation is completed. The execution is in system mode regardless of the `accessLevel` parameter. Use the
action callback to update org data according to the operation's results. The callback object must extend
`DataSource.AsyncDeleteCallback`.

**accessLevel**
Type: `System.AccessLevel`

(Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

**Return Value**
Type: `List<Database.DeleteResult>`

Status results for the delete operation. Each result corresponds to a record processed by this asynchronous operation and is associated with a unique identifier (asyncLocator). The `asyncLocator` value is included in the errors array of the result. You can retrieve this identifier with `Database.getAsyncLocator()`. Retrieve the final result with `Database.getAsyncDeleteResult()`.

**deleteAsync(sobject, callback, accessLevel)**

Initiates a request to delete the external data that corresponds to the specified external object record. The request is executed asynchronously, as a background operation, and is sent to the external system that's defined by the external object's associated external data source. Allows referencing a callback class whose `processDelete` method is called after deletion.

**Signature**

```java
public static Database.DeleteResult deleteAsync(SObject sobject, DataSource.AsyncDeleteCallback callback, System.AccessLevel accessLevel)
```

**Parameters**

- **sobject**
  Type: `SObject`
  The external object record to delete.

- **callback**
  Type: `DataSource.AsyncDeleteCallback`
  The callback that contains the state in the originating context and an action (the `processDelete` method) that is executed after the insert operation is completed. The execution is in system mode regardless of the `accessLevel` parameter. Use the action callback to update org data according to the operation's results. The callback object must extend `DataSource.AsyncDeleteCallback`.

- **accessLevel**
  Type: `System.AccessLevel`

  (Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.
Return Value
Type: Database.DeleteResult
Status result for the delete operation. The result corresponds to the record processed by this asynchronous operation and is associated with a unique identifier (asyncLocator). The asyncLocator value is included in the errors array of the result. You can retrieve this identifier with Database.getAsyncLocator(). Retrieve the final result with Database.getAsyncDeleteResult().

deleteAsync(sobjects, accessLevel)
Initiates requests to delete the external data that corresponds to the specified external object records. The requests are executed asynchronously, as background operations, and are sent to the external systems that are defined by the external objects’ associated external data sources.

Signature
public static List<Database.DeleteResult> deleteAsync(List<SObject> sobjects, System.AccessLevel accessLevel)

Parameters
sobjects
Type: List<SObject>
List of external object records to delete.
accessLevel
Type: System.AccessLevel
(Optional) The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value
Type: List<Database.DeleteResult>
Status results for the delete operation. Each result corresponds to a record processed by this asynchronous operation and is associated with a unique identifier (asyncLocator). The asyncLocator value is included in the errors array of the result. You can retrieve this identifier with Database.getAsyncLocator(). Retrieve the final result with Database.getAsyncDeleteResult().

deleteAsync(object, accessLevel)
Initiates a request to delete the external data that corresponds to the specified external object record. The request is executed asynchronously, as a background operation, and is sent to the external system that’s defined by the external object’s associated external data source.

Signature
public static Database.DeleteResult deleteAsync(SObject object, System.AccessLevel accessLevel)
Parameters

`sobject`
Type: `SObject`
The external object record to delete.

`accessLevel`
Type: `System.AccessLevel`
(Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value

Type: `Database.DeleteResult`
Status result for the delete operation. The result corresponds to the record processed by this asynchronous operation and is associated with a unique identifier (`asyncLocator`). The `asyncLocator` value is included in the errors array of the result. You can retrieve this identifier with `Database.getAsyncLocator()`. Retrieve the final result with `Database.getAsyncDeleteResult()`.

deleteImmediate(sobjects)

Initiates requests to delete the external data that corresponds to the specified external object records. The requests are executed synchronously and are sent to the external systems that are defined by the external objects’ associated external data sources. If the Apex transaction contains pending changes, the synchronous operations can’t be completed and throw exceptions.

Signature

`public static List<Database.DeleteResult> deleteImmediate(List<SObject> sobjects)`

Parameters

`sobjects`
Type: `List<SObject>`
List of external object records to delete.

Return Value

Type: `List<Database.DeleteResult>`
Status results for the delete operation.

Usage

The batch limit for big objects using `deleteImmediate()` is 50,000 records at once.

deleteImmediate(object)

Initiates a request to delete the external data that corresponds to the specified external object record. The request is executed synchronously and is sent to the external system that’s defined by the external object’s associated external data source. If the Apex transaction contains pending changes, the synchronous operation can’t be completed and throws an exception.
Signature

public static Database.DeleteResult deleteImmediate(SObject sobject)

Parameters

sobject
Type: SObject
The external object record to delete.

Return Value

Type: Database.DeleteResult
Status result for the delete operation.

databaseImmediate(sobjects, accessLevel)

Initiates requests to delete the external data that corresponds to the specified external object records. The requests are executed synchronously and are sent to the external systems that are defined by the external objects' associated external data sources. If the Apex transaction contains pending changes, the synchronous operations can't be completed and throw exceptions.

Signature

public static List<Database.DeleteResult> deleteImmediate(List<SObject> sobjects, System.AccessLevel accessLevel)

Parameters

sobjects
Type: List<SObject>
List of external object records to delete.

accessLevel
Type: System.AccessLevel
(Optional) The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value

Type: List<Database.DeleteResult>
Status results for the delete operation.

Usage

The batch limit for big objects using deleteImmediate() is 50,000 records at once.
**deleteImmediate(sobject, accessLevel)**

Initiates a request to delete the external data that corresponds to the specified external object record. The request is executed synchronously and is sent to the external system that’s defined by the external object’s associated external data source. If the Apex transaction contains pending changes, the synchronous operation can’t be completed and throws an exception.

**Signature**

```
public static Database.DeleteResult deleteImmediate(SObject sobject, System.AccessLevel accessLevel)
```

**Parameters**

- **sobject**
  - Type: SObject
  - The external object record to delete.

- **accessLevel**
  - Type: System.AccessLevel
  - (Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

**Return Value**

- Type: Database.DeleteResult
  - Status result for the delete operation.

**emptyRecycleBin(recordIds)**

Permanently deletes the specified records from the Recycle Bin.

**Signature**

```
public static Database.EmptyRecycleBinResult[] emptyRecycleBin(ID [] recordIds)
```

**Parameters**

- **recordIds**
  - Type: ID[]

**Return Value**

- Type: Database.EmptyRecycleBinResult[]

**Usage**

Note the following:

- After records are deleted using this method, they cannot be undeleted.
Empty Recycle Bin (obj)

Permanently deletes the specified sObject from the Recycle Bin.

Signature

public static Database.EmptyRecycleBinResult emptyRecycleBin(sObject obj)

Parameters

obj
Type: sObject

Return Value

Type: Database.EmptyRecycleBinResult

Usage

Note the following:

- After an sObject is deleted using this method, it cannot be undeleted.
- Only 10,000 sObjects can be specified for deletion.
- The logged-in user can delete any sObject (that can be queried) in their Recycle Bin, or the recycle bins of any subordinates. If the logged-in user has “Modify All Data” permission, they can query and delete sObjects from any Recycle Bin in the organization.
- Do not include an sObject that was deleted due to a cascade delete; otherwise an error occurs. For example, if an account is deleted, all related contacts, opportunities, contracts, and so on are also deleted. Only include sObjects of the top-level account. All related sObjects are automatically removed.

Empty Recycle Bin (listOfSObjects)

Permanently deletes the specified sObjects from the Recycle Bin.

Signature

public static Database.EmptyRecycleBinResult[] emptyRecycleBin(sObject[] listOfSObjects)

Parameters

listOfSObjects
Type: sObject[]
Return Value
Type: Database.EmptyRecycleBinResult[]

Usage
Note the following:
• After an sObject is deleted using this method, it cannot be undeleted.
• Only 10,000 sObjects can be specified for deletion.
• The logged-in user can delete any sObject (that can be queried) in their Recycle Bin, or the recycle bins of any subordinates. If the logged-in user has “Modify All Data” permission, they can query and delete sObjects from any Recycle Bin in the organization.
• Do not include an sObject that was deleted due to a cascade delete; otherwise an error occurs. For example, if an account is deleted, all related contacts, opportunities, contracts, and so on are also deleted. Only include sObjects of the top-level account. All related sObjects are automatically removed.

executeBatch (batchClassObject)
Submits a batch Apex job for execution corresponding to the specified class.

Signature
public static ID executeBatch(Object batchClassObject)

Parameters
batchClassObject
Type: Object
An instance of a class that implements the Database.Batchable interface.

Return Value
Type: ID
The ID of the new batch job (AsyncApexJob).

Usage
When calling this method, Salesforce chunks the records returned by the start method of the batch class into batches of 200, and then passes each batch to the execute method. Apex governor limits are reset for each execution of execute.
For more information, see Using Batch Apex.

Versioned Behavior Changes
If the executeBatch call fails to acquire an Apex flex queue lock:
• In API version 52.0 and later, the call throws a System.AsyncException.
• In API version 51.0 and earlier, the call returns an empty ID, “000000000000000”, instead of throwing an exception.
**executeBatch(batchClassObject, scope)**
Submits a batch Apex job for execution using the specified class and scope.

**Signature**
```
public static ID executeBatch(Object batchClassObject, Integer scope)
```

**Parameters**
- **batchClassObject**
  - Type: Object
  - An instance of a class that implements the Database.Batchable interface.
- **scope**
  - Type: Integer
  - Number of records to be passed into the execute method for batch processing.

**Return Value**
- Type: ID
- The ID of the new batch job (AsyncApexJob).

**Usage**
The value for `scope` must be greater than 0.

If the `start` method of the batch class returns a `Database.QueryLocator`, the scope parameter of `Database.executeBatch` can have a maximum value of 2,000. If set to a higher value, Salesforce chunks the records returned by the `QueryLocator` into smaller batches of up to 200 records. If the `start` method of the batch class returns an iterable, the scope parameter value has no upper limit; however, if you use a very high number, you could run into other limits.

Apex governor limits are reset for each execution of `execute`.

For more information, see Using Batch Apex.

**Versioned Behavior Changes**
If the `executeBatch` call fails to acquire an Apex flex queue lock:
- In API version 52.0 and later, the call throws a `System.AsyncException`.
- In API version 51.0 and earlier, the call returns an empty ID, "000000000000000", instead of throwing an exception.

**getAsyncDeleteResult(deleteResult)**
Retrieves the status of an asynchronous delete operation that’s identified by a `Database.DeleteResult` object.

**Signature**
```
public static Database.DeleteResult getAsyncDeleteResult(Database.DeleteResult deleteResult)
```
Parameters

deleteResult
   Type: Database.DeleteResult
   The result record for the delete operation being retrieved.

Return Value

Type: Database.DeleteResult
The result of a completed asynchronous delete of a record or records.

getAsyncDeleteResult(asyncLocator)

Retrieves the result of an asynchronous delete operation based on the result’s unique identifier.

Signature

public static Database.DeleteResult getAsyncDeleteResult(String asyncLocator)

Parameters

asyncLocator
   Type: String
   The unique identifier associated with the result of an asynchronous operation.

Return Value

Type: Database.DeleteResult
The result of a completed asynchronous delete of a record or records.

getAsyncLocator(result)

Returns the asyncLocator associated with the result of a specified asynchronous insert, update, or delete operation.

Signature

public static String getAsyncLocator(Object result)

Parameters

result
   Type: Object
   The saved result of an asynchronous insert, update, or delete operation. The result object can be of type Database.SaveResult or Database.DeleteResult.

Return Value

Type: String
The unique identifier associated with the result of the specified operation.
getAsyncSaveResult (saveResult)
Returns the status of an asynchronous insert or update operation that’s identified by a Database.SaveResult object.

Signature
public static Database.SaveResult getAsyncSaveResult(Database.SaveResult saveResult)

Parameters
saveResult
Type: Database.SaveResult
The result record for the insert or update operation being retrieved.

Return Value
Database.SaveResult
The result of a completed asynchronous operation on a record or records.

getAsyncSaveResult (asyncLocator)
Returns the status of an asynchronous insert or update operation based on the unique identifier associated with each modification.

Signature
public static Database.SaveResult getAsyncSaveResult(String asyncLocator)

Parameters
asyncLocator
Type: String
The unique identifier associated with the result of an asynchronous operation.

Return Value
Database.SaveResult
The result of a completed asynchronous operation on a record or records.

getDeleted(sObjectType, startDate, endDate)
Returns the list of individual records that have been deleted for an sObject type within the specified start and end dates and times and that are still in the Recycle Bin.

Signature
public static Database.GetDeletedResult getDeleted(String sObjectType, Datetime startDate, Datetime endDate)
Parameters

sObjectType
Type: String
The sObjectType argument is the sObject type name for which to get the deleted records, such as account or merchandise__c.

startDate
Type: Datetime
Start date and time of the deleted records time window.

endDate
Type: Datetime
End date and time of the deleted records time window.

Return Value
Type: Database.GetDeletedResult

Usage
Because the Recycle Bin holds records up to 15 days, results are returned for no more than 15 days previous to the day the call is executed (or earlier if an administrator has purged the Recycle Bin).

Example
```
Database.GetDeletedResult r =
    Database.getDeleted(
        'Merchandise__c',
        Datetime.now().addHours(-1),
        Datetime.now());
```

getQueryLocator(listOfQueries)
Creates a QueryLocator object used in batch Apex or Visualforce.

Signature
```
public static Database.QueryLocator getQueryLocator(sObject [] listOfQueries)
```

Parameters

listOfQueries
Type: sObject []

Return Value
Type: Database.QueryLocator

Usage
You can't use getQueryLocator with any query that contains an aggregate function.
Each executed `getQueryLocator` method counts against the governor limit of 10,000 total records retrieved and the total number of SOQL queries issued.

For more information, see Understanding Apex Managed Sharing, and `IdeaStandardSetController` Class.

**getQueryLocator**(query)

Creates a QueryLocator object used in batch Apex or Visualforce.

**Signature**

```java
public static Database.QueryLocator getQueryLocator(String query)
```

**Parameters**

- **query**
  - Type: String

**Return Value**

Type: Database.QueryLocator

**Usage**

You can’t use `getQueryLocator` with any query that contains an aggregate function.

Each executed `getQueryLocator` method counts against the governor limit of 10,000 total records retrieved and the total number of SOQL queries issued.

For more information, see Understanding Apex Managed Sharing, and `StandardSetController` Class.

**getQueryLocator**(listOfQueries, accessLevel)

Creates a QueryLocator object used in batch Apex or Visualforce.

**Signature**

```java
public static Database.QueryLocator getQueryLocator(sObject [] listofQueries, System.AccessLevel accessLevel)
```

**Parameters**

- **listOfQueries**
  - Type: sObject []

- **accessLevel**
  - Type: System.AccessLevel

(Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.
Return Value
Type: Database.QueryLocator

Usage
The access level is evaluated only when the QueryLocator is created. A QueryLocator can be long lived, such as when used in a batch. We don’t reevaluate the object and field-level security with each iteration of the QueryLocator. As a result, if you specify user mode, and then change the security settings after the QueryLocator is created, the new settings aren’t enforced.

You can’t use getQueryLocator with any query that contains an aggregate function.

Each executed getQueryLocator method counts against the governor limit of 10,000 total records retrieved and the total number of SOQL queries issued.

For more information, see Understanding Apex Managed Sharing, and IdeaStandardSetController Class.

ggetQueryLocator(query, accessLevel)
Creates a QueryLocator object used in batch Apex or Visualforce.

Signature
public static Database.QueryLocator getQueryLocator(String query, System.AccessLevel accessLevel)

Parameters
query
  Type: String
accessLevel
  Type: System.AccessLevel
    (Optional) The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value
Type: Database.QueryLocator

Usage
The access level is evaluated only when the QueryLocator is created. A QueryLocator can be long lived, such as when used in a batch. We don’t reevaluate the object and field-level security with each iteration of the QueryLocator. As a result, if you specify user mode, and then change the security settings after the QueryLocator is created, the new settings aren’t enforced.

You can’t use getQueryLocator with any query that contains an aggregate function.

Each executed getQueryLocator method counts against the governor limit of 10,000 total records retrieved and the total number of SOQL queries issued.

For more information, see Understanding Apex Managed Sharing, and StandardSetController Class.
**getQueryLocatorWithBinds** *(query, bindMap, accessLevel)*

Creates a QueryLocator object used in batch Apex or Visualforce. Bind variables in the query are resolved from the bindMap Map parameter directly with the key, rather than from Apex code variables.

**Signature**

```java
public static Database.QueryLocator getQueryLocatorWithBinds(String query, Map<String, Object> bindMap, System.AccessLevel accessLevel)
```

**Parameters**

- **query**
  - Type: **String**
  - SOQL query that includes Apex bind variables preceded by a colon. All bind variables must have a key in the bindMap Map.

- **bindMap**
  - Type: **Map<String, Object>**
  - A map that contains keys for each bind variable specified in the SOQL queryString and its value. The keys can’t be null or duplicates, and the values can’t be null or empty strings.

- **accessLevel**
  - Type: **System.AccessLevel**
  - The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced.

**Return Value**

- Type: **Database.QueryLocator**

**Usage**

The access level is evaluated only when the QueryLocator is created. A QueryLocator can be long lived, such as when used in a batch. We don’t reevaluate the object and field-level security with each iteration of the QueryLocator. As a result, if you specify user mode, and then change the security settings after the QueryLocator is created, the new settings aren’t enforced.

You can’t use `getQueryLocatorWithBinds` with any query that contains an aggregate function.

Each executed `getQueryLocatorWithBinds` method counts against the governor limit for the total number of records retrieved by Database.getQueryLocator(10,000) and the total number of SOQL queries issued. See Per Transaction Apex Limits.

For more information, see Understanding Apex Managed Sharing, and StandardSetController Class.

**Example**

In this example, the SOQL query uses a bind variable for an Account name. Its value (Acme Corporation) is passed in using the acctBinds Map.

```java
public class TestBatch implements Database.Batchable<sObject>{

    private Map<String, Object> acctBinds = new Map<String, Object>{'acctName' => 'Acme
```
Corporation'};

    private String query = 'Select Id From Account where name = :acctName';

    public Database.QueryLocator start(Database.BatchableContext BC)
    {
        return Database.getQueryLocatorWithBinds(query, acctBinds, AccessLevel.USER_MODE);
    }

    public void execute(Database.BatchableContext BC, List<sObject> scope)
    {
    }

    public void finish(Database.BatchableContext BC)
    {
    }

getUpdated( objectType, startDate, endDate)

Returns the list of individual records that have been updated for an sObject type within the specified start and end dates and times.

Signature

public static Database.GetUpdatedResult getUpdated(String objectType, Datetime startDate, Datetime endDate)

Parameters

objectType
    Type: String
    The objectType argument is the sObject type name for which to get the updated records, such as account or merchandise__c.

startDate
    Type: Datetime
    The startDate argument is the start date and time of the updated records time window.

endDate
    Type: Datetime
    The endDate argument is the end date and time of the updated records time window.

Return Value

Type: Database.GetUpdatedResult

Usage

The date range for the returned results is no more than 30 days previous to the day the call is executed.

Example

Database.GetUpdatedResult r =
    Database.getUpdated(
        'Merchandise__c',
    )
**insert(recordToInsert, allOrNone)**

Adds an sObject, such as an individual account or contact, to your organization’s data.

**Signature**

```java
public static Database.SaveResult insert(sObject recordToInsert, Boolean allOrNone)
```

**Parameters**

- `recordToInsert`
  - Type: `sObject`
- `allOrNone`
  - Type: `Boolean`

The optional `allOrNone` parameter specifies whether the operation allows partial success. If you specify `false` for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set `true`, an exception is thrown if the method is not successful.

**Return Value**

Type: `Database.SaveResult`

**Usage**

`insert` is analogous to the INSERT statement in SQL.

Apex classes and triggers saved (compiled) using API version 15.0 and higher produce a runtime error if you assign a String value that is too long for the field.

Each executed `insert` method counts against the governor limit for DML statements.

**insert(recordsToInsert, allOrNone)**

Adds one or more sObjects, such as individual accounts or contacts, to your organization’s data.

**Signature**

```java
public static Database.SaveResult[] insert(sObject[] recordsToInsert, Boolean allOrNone)
```

**Parameters**

- `recordsToInsert`
  - Type: `sObject[]`
- `allOrNone`
  - Type: `Boolean`
The optional `allOrNone` parameter specifies whether the operation allows partial success. If you specify `false` for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter isn’t set or is set `true`, an exception is thrown if the method isn’t successful. If the parameter is specified as `false` and a before-trigger assigns an invalid value to a field, the partial set of valid records isn’t inserted.

**Return Value**

Type: `Database.SaveResult[]`

**Usage**

`insert` is analogous to the `INSERT` statement in SQL.

Apex classes and triggers saved (compiled) using API version 15.0 and higher produce a runtime error if you assign a String value that is too long for the field.

Each executed `insert` method counts against the governor limit for DML statements.

**Example**

Example:

The following example inserts two accounts:

```java
Account a = new Account(name = 'Acme1');
Database.SaveResult[] lsr = Database.insert(
    new Account[]{a, new Account(Name = 'Acme2')},
    false);
```

**insert(recordToInsert, dmlOptions)**

Adds an sObject, such as an individual account or contact, to your organization’s data.

**Signature**

```java
public static Database.SaveResult insert(sObject recordToInsert, Database.DMLOptions dmlOptions)
```

**Parameters**

`recordToInsert`

Type: `sObject`

`dmlOptions`

Type: `Database.DMLOptions`

The optional `dmlOptions` parameter specifies additional data for the transaction, such as assignment rule information or rollback behavior when errors occur during record insertions.

**Return Value**

Type: `Database.SaveResult`
Usage

*insert* is analogous to the INSERT statement in SQL.

Apex classes and triggers saved (compiled) using API version 15.0 and higher produce a runtime error if you assign a String value that is too long for the field.

Each executed *insert* method counts against the governor limit for DML statements.

**insert(recordsToInsert, dmlOptions)**

Adds one or more sObjects, such as individual accounts or contacts, to your organization’s data.

**Signature**

```java
public static Database.SaveResult insert(sObject[] recordsToInsert, Database.DMLOptions dmlOptions)
```

**Parameters**

- `recordsToInsert`
  
  Type: `sObject[]`

- `dmlOptions`
  
  Type: `Database.DMLOptions`

  The optional `dmlOptions` parameter specifies additional data for the transaction, such as assignment rule information or rollback behavior when errors occur during record insertions.

**Return Value**

Type: `Database.SaveResult[]`

Usage

*insert* is analogous to the INSERT statement in SQL.

Apex classes and triggers saved (compiled) using API version 15.0 and higher produce a runtime error if you assign a String value that is too long for the field.

Each executed *insert* method counts against the governor limit for DML statements.

**insert(recordToInsert, allOrNone, accessLevel)**

Adds an sObject, such as an individual account or contact, to your organization’s data.

**Signature**

```java
public static Database.SaveResult insert(SObject recordToInsert, Boolean allOrNone, System.AccessLevel accessLevel)
```

**Parameters**

- `recordToInsert`
  
  Type: `sObject`
**allOrNone**

Type: Boolean

The optional `allOrNone` parameter specifies whether the operation allows partial success. If you specify `false` for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set `true`, an exception is thrown if the method is not successful.

**accessLevel**

Type: System.AccessLevel

(Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

**Return Value**

Type: Database.SaveResult

**Usage**

If you use the `accessLevel` parameter to specify that the method runs in user mode, we report all encountered inaccessible fields. The way to retrieve the names of these inaccessible fields depends on the value of this method’s `allOrNone` parameter, or the equivalent `DmlOptions.optAllOrNone` property. If you specify that:

- `allOrNone=true` or `DmlOptions.optAllOrNone=true`: Catch the `DMLException` and use the `DMLException.getDMLFieldNames()` method to retrieve the list of inaccessible fields. See Exception Class and Built-In Exceptions for more information.

- `allOrNone=false` or `DmlOptions.optAllOrNone=false`: For each failing record, we update the `Database.Error` object that results from the DML operation. Use the `Error.getFields()` method to retrieve the list of inaccessible fields. See the Error Class methods for more information.

**insert** is analogous to the INSERT statement in SQL.

Apex classes and triggers saved (compiled) using API version 15.0 and higher produce a runtime error if you assign a String value that is too long for the field.

Each executed `insert` method counts against the governor limit for DML statements.

**insert(recordsToInsert, allOrNone, accessLevel)**

Adds one or more sObjects, such as individual accounts or contacts, to your organization's data.

**Signature**

```java
public static List<Database.SaveResult> insert(List<SObject> recordsToInsert, Boolean allOrNone, System.AccessLevel accessLevel)
```

**Parameters**

**recordsToInsert**

Type: List<sObject>
allOrNone
Type: Boolean
The optional allOrNone parameter specifies whether the operation allows partial success. If you specify false for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter isn’t set or is set true, an exception is thrown if the method isn’t successful. If the parameter is specified as false and a before-trigger assigns an invalid value to a field, the partial set of valid records isn’t inserted.

accessLevel
Type: System.AccessLevel
(Optional) The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value
Type: List<Database.SaveResult>

Usage
If you use the accessLevel parameter to specify that the method runs in user mode, we report all encountered inaccessible fields. The way to retrieve the names of these inaccessible fields depends on the value of this method’s allOrNone parameter, or the equivalent DmlOptions.optAllOrNone property. If you specify that:

• allOrNone=true or DmlOptions.optAllOrNone=true: Catch the DMLException and use the DMLException.getDMLFieldNames() method to retrieve the list of inaccessible fields. See Exception Class and Built-In Exceptions for more information.
• allOrNone=false or DmlOptions.optAllOrNone=false: For each failing record, we update the Database.Error object that results from the DML operation. Use the Error.getFields() method to retrieve the list of inaccessible fields. See the Error Class methods for more information.

insert is analogous to the INSERT statement in SQL.

Apex classes and triggers saved (compiled) using API version 15.0 and higher produce a runtime error if you assign a String value that is too long for the field.

Each executed insert method counts against the governor limit for DML statements.

insert(recordToInsert, dmlOptions, accessLevel)
Adds an sObject, such as an individual account or contact, to your organization’s data.

Signature
public static Database.SaveResult insert(SObject recordToInsert, Database.DMLOptions dmlOptions, System.AccessLevel accessLevel)

Parameters
recordToInsert
Type: sObject
**dmlOptions**
Type: `Database.DMLOptions`

The optional `dmlOptions` parameter specifies additional data for the transaction, such as assignment rule information or rollback behavior when errors occur during record insertions.

**accessLevel**
Type: `System.AccessLevel`

(Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

**Return Value**
Type: `Database.SaveResult`

**Usage**
`insert` is analogous to the INSERT statement in SQL.

Apex classes and triggers saved (compiled) using API version 15.0 and higher produce a runtime error if you assign a String value that is too long for the field.

Each executed `insert` method counts against the governor limit for DML statements.

\[\text{insert(recordsToInsert, dmlOptions, accessLevel)}\]

Adds one or more sObjects, such as individual accounts or contacts, to your organization's data.

**Signature**

```java
public static List<Database.SaveResult> insert(List<SObject> recordsToInsert, Database.DMLOptions dmlOptions, System.AccessLevel accessLevel)
```

**Parameters**

- **recordsToInsert**
  Type: `List<SObject>`

- **dmlOptions**
  Type: `Database.DMLOptions`

  The optional `dmlOptions` parameter specifies additional data for the transaction, such as assignment rule information or rollback behavior when errors occur during record insertions.

- **accessLevel**
  Type: `System.AccessLevel`

  (Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.
Return Value
Type: List<Database.SaveResult>

Usage
insert is analogous to the INSERT statement in SQL.
Apex classes and triggers saved (compiled) using API version 15.0 and higher produce a runtime error if you assign a String value that is too long for the field.
Each executed insert method counts against the governor limit for DML statements.

insertAsync(sobjects, callback)
Initiates requests to add external object data to the relevant external systems. The requests are executed asynchronously, as background operations, and are sent to the external systems that are defined by the external objects’ associated external data sources. Allows referencing a callback class whose processSave method is called for each record after the remote operations are completed.

Signature
public static List<Database.SaveResult> insertAsync(List<SObject> sobjects, DataSource.AsyncSaveCallback callback)

Parameters
sobjects
Type: List<SObject>
List of external object records to insert.
callback
Type: DataSource.AsyncSaveCallback
The callback object that contains the state in the originating context and an action (the processSave method) that executes after the insert operation is completed. Use the action callback to update org data according to the operation’s results. The callback object must extend DataSource.AsyncSaveCallback.

Return Value
Type: List<Database.SaveResult>
Status results for the insert operation. Each result corresponds to a record processed by this asynchronous operation and is associated with a unique identifier (asyncLocator). The asyncLocator value is included in the errors array of the result. You can retrieve this identifier with Database.getAsyncLocator(). Retrieve the final result with Database.getAsyncSaveResult().

Usage
Database.insertAsync() methods can't be executed in the context of a portal user, even when the portal user is a community member. To add external object records via Apex, use Database.insertImmediate() methods.
**insertAsync(object, callback)**

Initiates a request to add external object data to the relevant external system. The request is executed asynchronously, as a background operation, and is sent to the external system that's defined by the external object's associated external data source. Allows referencing a callback class whose `processSave` method is called after the remote operation is completed.

**Signature**

```java
public static Database.SaveResult insertAsync(SObject object,
DataSource.AsyncSaveCallback callback)
```

**Parameters**

- `object`
  - Type: `SObject`
  - The external object record to insert.

- `callback`
  - Type: `DataSource.AsyncSaveCallback`
  - The callback object that contains the state in the originating context and an action (the `processSave` method) that executes after the insert operation is completed. Use the action callback to update org data according to the operation's results. The callback object must extend `DataSource.AsyncSaveCallback`.

**Return Value**

Type: `Database.SaveResult`

Status result for the insert operation. The result corresponds to the record processed by this asynchronous operation and is associated with a unique identifier (`asyncLocator`). The `asyncLocator` value is included in the errors array of the result. You can retrieve this identifier with `Database.getAsyncLocator()`. Retrieve the final result with `Database.getAsyncSaveResult()`.

**Usage**

`Database.insertAsync()` methods can't be executed in the context of a portal user, even when the portal user is a community member. To add external object records via Apex, use `Database.insertImmediate()` methods.

**insertAsync(objects)**

Initiates requests to add external object data to the relevant external systems. The requests are executed asynchronously, as background operations, and are sent to the external systems that are defined by the external objects' associated external data sources.

**Signature**

```java
public static List<Database.SaveResult> insertAsync(List<SObject> objects)
```

**Parameters**

- `objects`
  - Type: `List<SObject>`
  - List of external object records to insert.
Return Value

Type: List<Database.SaveResult>

Status results for the insert operation. Each result corresponds to a record processed by this asynchronous operation and is associated with a unique identifier (asyncLocator). The asyncLocator value is included in the errors array of the result. You can retrieve this identifier with Database.getAsyncLocator(). Retrieve the final result with Database.getAsyncSaveResult().

Usage

Database.insertAsync() methods can’t be executed in the context of a portal user, even when the portal user is a community member. To add external object records via Apex, use Database.insertImmediate() methods.

insertAsync(sobject)

Initiates a request to add external object data to the relevant external system. The request is executed asynchronously, as a background operation, and is sent to the external system that’s defined by the external object’s associated external data source.

Signature

public static Database.SaveResult insertAsync(SObject object)

Parameters

object

Type: SObject

The external object record to insert.

Return Value

Type: Database.SaveResult

Status result for the insert operation. The result corresponds to the record processed by this asynchronous operation and is associated with a unique identifier (asyncLocator). The asyncLocator value is included in the errors array of the result. You can retrieve this identifier with Database.getAsyncLocator(). Retrieve the final result with Database.getAsyncSaveResult().

Usage

Database.insertAsync() methods can’t be executed in the context of a portal user, even when the portal user is a community member. To add external object records via Apex, use Database.insertImmediate() methods.

insertAsync(sobjects, callback, accessLevel)

Initiates requests to add external object data to the relevant external systems. The requests are executed asynchronously, as background operations, and are sent to the external systems that are defined by the external objects’ associated external data sources. Allows referencing a callback class whose processSave method is called for each record after the remote operations are completed.

Signature

public static List<Database.SaveResult> insertAsync(List<SObject> objects, DataSource.AsyncSaveCallback callback, System.AccessLevel accessLevel)
Parameters>

`sobjects`  
Type: List<SObject>  
List of external object records to insert.

`callback`  
Type: DataSource.AsyncSaveCallback  
The callback object that contains the state in the originating context and an action (the `processSave` method) that executes after the insert operation is completed. The execution is in system mode regardless of the `accessLevel` parameter. Use the action callback to update org data according to the operation's results. The callback object must extend `DataSource.AsyncSaveCallback`.

`accessLevel`  
Type: System.AccessLevel  
(Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value>

Type: List<Database.SaveResult>  
Status results for the insert operation. Each result corresponds to a record processed by this asynchronous operation and is associated with a unique identifier (`asyncLocator`). The `asyncLocator` value is included in the errors array of the result. You can retrieve this identifier with `Database.getAsyncLocator()`. Retrieve the final result with `Database.getAsyncSaveResult()`.

Usage>

`Database.insertAsync()` methods can't be executed in the context of a portal user, even when the portal user is a community member. To add external object records via Apex, use `Database.insertImmediate()` methods.

`insertAsync(sobject, callback, accessLevel)`  
Initiates a request to add external object data to the relevant external system. The request is executed asynchronously, as a background operation, and is sent to the external system that's defined by the external object's associated external data source. Allows referencing a callback class whose `processSave` method is called after the remote operation is completed.

Signature  

```java
public static Database.SaveResult insertAsync(SObject sobject,  
DataSource.AsyncSaveCallback callback, System.AccessLevel accessLevel)
```

Parameters  

`sobject`  
Type: SObject  
The external object record to insert.

`callback`  
Type: DataSource.AsyncSaveCallback
The callback object that contains the state in the originating context and an action (the processSave method) that executes after the insert operation is completed. The execution is in system mode regardless of the accessLevel parameter. Use the action callback to update org data according to the operation's results. The callback object must extend DataSource.AsyncSaveCallback.

accessLevel
Type: System.AccessLevel
(Optional) The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value
Type: Database.SaveResult
Status result for the insert operation. The result corresponds to the record processed by this asynchronous operation and is associated with a unique identifier (asyncLocator). The asyncLocator value is included in the errors array of the result. You can retrieve this identifier with Database.getAsyncLocator(). Retrieve the final result with Database.getAsyncSaveResult().

Usage
Database.insertAsync() methods can’t be executed in the context of a portal user, even when the portal user is a community member. To add external object records via Apex, use Database.insertImmediate() methods.

insertAsync(sobjects, accessLevel)
Initiates requests to add external object data to the relevant external systems. The requests are executed asynchronously, as background operations, and are sent to the external systems that are defined by the external objects’ associated external data sources.

Signature
public static List<Database.SaveResult> insertAsync(List<SObject> sobjects, System.AccessLevel accessLevel)

Parameters
sobjects
Type: List<SObject>
List of external object records to insert.

accessLevel
Type: System.AccessLevel
(Optional) The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value
Type: List<Database.SaveResult>
Status results for the insert operation. Each result corresponds to a record processed by this asynchronous operation and is associated with a unique identifier (asyncLocator). The asyncLocator value is included in the errors array of the result. You can retrieve this identifier with Database.getAsyncLocator(). Retrieve the final result with Database.getAsyncSaveResult().

Usage

Database.insertAsync() methods can't be executed in the context of a portal user, even when the portal user is a community member. To add external object records via Apex, use Database.insertImmediate() methods.

insertAsync(object, accessLevel)

Initiates a request to add external object data to the relevant external system. The request is executed asynchronously, as a background operation, and is sent to the external system that's defined by the external object's associated external data source.

Signature

public static Database.SaveResult insertAsync(SObject object, System.AccessLevel accessLevel)

Parameters

object

Type: SObject

The external object record to insert.

accessLevel

Type: System.AccessLevel

(Optional) The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value

Type: Database.SaveResult

Status result for the insert operation. The result corresponds to the record processed by this asynchronous operation and is associated with a unique identifier (asyncLocator). The asyncLocator value is included in the errors array of the result. You can retrieve this identifier with Database.getAsyncLocator(). Retrieve the final result with Database.getAsyncSaveResult().

Usage

Database.insertAsync() methods can't be executed in the context of a portal user, even when the portal user is a community member. To add external object records via Apex, use Database.insertImmediate() methods.

insertImmediate(objects)

Initiates requests to add external object data to the relevant external systems. The requests are executed synchronously and are sent to the external systems that are defined by the external objects' associated external data sources. If the Apex transaction contains pending changes, the synchronous operations can't be completed and throw exceptions.
public static List<Database.SaveResult> insertImmediate(List<SObject> objects)

Parameters
objects
  Type: List<SObject>
  List of external object records to insert.

Return Value
Type: List<Database.SaveResult>
Status results for the insert operation.

Usage
The operation allows partial success. If one or more record inserts fail, the method doesn’t throw an exception and the remainder of the DML operation can still succeed. The returned SaveResult objects indicate whether the operation was successful. If it wasn’t successful, the objects also return the error code and description.

insertImmediate(object)
Initiates a request to add external object data to the relevant external system. The request is executed synchronously and is sent to the external system that’s defined by the external object’s associated external data source. If the Apex transaction contains pending changes, the synchronous operation can’t be completed and throws an exception.

public static Database.SaveResult insertImmediate(SObject object)

Parameters
object
  Type: SObject
  The external object record to insert.

Return Value
Type: Database.SaveResult
Status result for the insert operation.

Usage
If a record insert fails, the method doesn’t throw an exception. The returned SaveResult object indicates whether the operation was successful. If it wasn’t successful, the object returns the error code and description.
**insertImmediate**(*objects*, *accessLevel*)

Initiates requests to add external object data to the relevant external systems. The requests are executed synchronously and are sent to the external systems that are defined by the external objects’ associated external data sources. If the Apex transaction contains pending changes, the synchronous operations can’t be completed and throw exceptions.

**Signature**

```java
public static List<Database.SaveResult> insertImmediate(List<SObject> objects, System.AccessLevel accessLevel)
```

**Parameters**

- **objects**
  - Type: List<SObject>
  - List of external object records to insert.

- **accessLevel**
  - Type: System.AccessLevel
  - (Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

**Return Value**

- Type: List<Database.SaveResult>
- Status results for the insert operation.

**Usage**

The operation allows partial success. If one or more record inserts fail, the method doesn’t throw an exception and the remainder of the DML operation can still succeed. The returned `SaveResult` objects indicate whether the operation was successful. If it wasn’t successful, the objects also return the error code and description.

**insertImmediate**(*object*, *accessLevel*)

Initiates a request to add external object data to the relevant external system. The request is executed synchronously and is sent to the external system that’s defined by the external object’s associated external data source. If the Apex transaction contains pending changes, the synchronous operation can’t be completed and throws an exception.

**Signature**

```java
public static Database.SaveResult insertImmediate(SObject object, System.AccessLevel accessLevel)
```

**Parameters**

- **object**
  - Type: SObject
The external object record to insert.

`accessLevel`
Type: `System.AccessLevel`  
(Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

**Return Value**
Type: `Database.SaveResult`  
Status result for the insert operation.

**Usage**
If a record update fails, the method doesn’t throw an exception. The returned `SaveResult` object indicates whether the operation was successful. If it failed, the object returns the error code and description.

**merge (masterRecord, duplicateId)**
Merges the specified duplicate record into the master sObject record of the same type, deleting the duplicate, and reparenting any related records. Merges only accounts, contacts, or leads.

**Signature**
`public static Database.MergeResult merge(sObject masterRecord, Id duplicateId)`

**Parameters**
- `masterRecord`
  Type: `sObject`  
  The master sObject record the duplicate record is merged into.
- `duplicateId`
  Type: `ID`  
  The ID of the record to merge with the master. This record must be of the same sObject type as the master.

**Return Value**
Type: `Database.MergeResult`  

**Usage**
Each executed `merge` method counts against the governor limit for DML statements.

**merge (masterRecord, duplicateRecord)**
Merges the specified duplicate sObject record into the master sObject of the same type, deleting the duplicate, and reparenting any related records.
Signature
`public static Database.MergeResult merge(sObject masterRecord, sObject duplicateRecord)`

Parameters
- `masterRecord`
  - Type: `sObject`
  - The master `sObject` record the duplicate record is merged into.
- `duplicateRecord`
  - Type: `sObject`
  - The `sObject` record to merge with the master. This `sObject` must be of the same type as the master.

Return Value
Type: `Database.MergeResult`

Usage
Each executed `merge` method counts against the governor limit for DML statements.

`merge(masterRecord, duplicateIds)`
Merges up to two records of the same `sObject` type into the master `sObject` record, deleting the others, and reparenting any related records.

Signature
`public static List<Database.MergeResult> merge(sObject masterRecord, List<Id> duplicateIds)`

Parameters
- `masterRecord`
  - Type: `sObject`
  - The master `sObject` record the other records are merged into.
- `duplicateIds`
  - Type: `List<Id>`
  - A list of `Ids` of up to two records to merge with the master. These records must be of the same `sObject` type as the master.

Return Value
Type: `List<Database.MergeResult>`

Usage
Each executed `merge` method counts against the governor limit for DML statements.
**merge (masterRecord, duplicateRecords)**

Merges up to two records of the same object type into the master sObject record, deleting the others, and reparenting any related records.

**Signature**

```java
public static List<Database.MergeResult> merge(sObject masterRecord, List<sObject> duplicateRecords)
```

**Parameters**

- **masterRecord**
  - Type: `sObject`
  - The master sObject record the other sObjects are merged into.

- **duplicateRecords**
  - Type: `List<sObject>`
  - A list of up to two sObject records to merge with the master. These sObjects must be of the same type as the master.

**Return Value**

Type: `List<Database.MergeResult>`

**Usage**

Each executed `merge` method counts against the governor limit for DML statements.

**merge (masterRecord, duplicateId, allOrNone)**

Merges the specified duplicate record into the master sObject record of the same type, optionally returning errors, if any, deleting the duplicate, and reparenting any related records. Merges only accounts, contacts, or leads.

**Signature**

```java
public static Database.MergeResult merge(sObject masterRecord, Id duplicateId, Boolean allOrNone)
```

**Parameters**

- **masterRecord**
  - Type: `sObject`
  - The master sObject record the duplicate record is merged into.

- **duplicateId**
  - Type: `ID`
  - The ID of the record to merge with the master. This record must be of the same sObject type as the master.

- **allOrNone**
  - Type: `Boolean`
Set to `false` to return any errors encountered in this operation as part of the returned result. If set to `true`, this method throws an exception if the operation fails. The default is `true`.

**Return Value**
Type: `Database.MergeResult`

**Usage**
Each executed `merge` method counts against the governor limit for DML statements.

**merge (masterRecord, duplicateRecord, allOrNone)**
Merges the specified duplicate sObject record into the master sObject of the same type, optionally returning errors, if any, deleting the duplicate, and reparenting any related records.

**Signature**
```
public static Database.MergeResult merge(sObject masterRecord, sObject duplicateRecord, Boolean allOrNone)
```

**Parameters**
- `masterRecord`
  Type: `sObject`
  The master sObject record the duplicate record is merged into.
- `duplicateRecord`
  Type: `sObject`
  The sObject record to merge with the master. This sObject must be of the same type as the master.
- `allOrNone`
  Type: `Boolean`
  Set to `false` to return any errors encountered in this operation as part of the returned result. If set to `true`, this method throws an exception if the operation fails. The default is `true`.

**Return Value**
Type: `Database.MergeResult`

**Usage**
Each executed `merge` method counts against the governor limit for DML statements.

**merge (masterRecord, duplicateIds, allOrNone)**
Merges up to two records of the same sObject type into the master sObject record, optionally returning errors, if any, deleting the duplicates, and reparenting any related records.
**Signature**

```
public static List<Database.MergeResult> merge(sObject masterRecord, List<Id> duplicateIds, Boolean allOrNone)
```

**Parameters**

- **masterRecord**
  - Type: `sObject`
  - The master `sObject` record the other records are merged into.

- **duplicateIds**
  - Type: `List<Id>`
  - A list of IDs of up to two records to merge with the master. These records must be of the same `sObject` type as the master.

- **allOrNone**
  - Type: `Boolean`
  - Set to `false` to return any errors encountered in this operation as part of the returned result. If set to `true`, this method throws an exception if the operation fails. The default is `true`.

**Return Value**

Type: `List<Database.MergeResult>`

**Usage**

Each executed `merge` method counts against the governor limit for DML statements.

**merge (masterRecord, duplicateRecords, allOrNone)**

Merges up to two records of the same object type into the master `sObject` record, optionally returning errors, if any, deleting the duplicates, and reparenting any related records.

**Signature**

```
public static List<Database.MergeResult> merge(sObject masterRecord, List<sObject> duplicateRecords, Boolean allOrNone)
```

**Parameters**

- **masterRecord**
  - Type: `sObject`
  - The master `sObject` record the other `sObjects` are merged into.

- **duplicateRecords**
  - Type: `List<sObject>`
  - A list of up to two `sObject` records to merge with the master. These `sObjects` must be of the same type as the master.

- **allOrNone**
  - Type: `Boolean`
Set to `false` to return any errors encountered in this operation as part of the returned result. If set to `true`, this method throws an exception if the operation fails. The default is `true`.

**Return Value**
Type: `List<Database.MergeResult>`

**Usage**
Each executed `merge` method counts against the governor limit for DML statements.

**merge(masterRecord, duplicateId, accessLevel)**
Merges the specified duplicate record into the master sObject record of the same type, deleting the duplicate, and reparenting any related records. Merges only accounts, contacts, or leads.

**Signature**

```java
public static Database.MergeResult merge(SObject masterRecord, Id duplicateId, System.AccessLevel accessLevel)
```

**Parameters**

- `masterRecord`
  Type: `sObject`
  The master sObject record the duplicate record is merged into.

- `duplicateId`
  Type: `ID`
  The ID of the record to merge with the master. This record must be of the same sObject type as the master.

- `accessLevel`
  Type: `System.AccessLevel`
  (Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

**Return Value**
Type: `Database.MergeResult`

**Usage**
Each executed `merge` method counts against the governor limit for DML statements.

**merge(masterRecord, duplicateRecord, accessLevel)**
Merges the specified duplicate sObject record into the master sObject of the same type, deleting the duplicate, and reparenting any related records.
Signature

public static Database.MergeResult merge(SObject masterRecord, SObject duplicateRecord, System.AccessLevel accessLevel)

Parameters

masterRecord
Type: SObject
The master sObject record the duplicate record is merged into.

duplicateRecord
Type: SObject
The sObject record to merge with the master. This sObject must be of the same type as the master.

accessLevel
Type: System.AccessLevel
(Optional) The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value

Type: Database.MergeResult

Usage

Each executed merge method counts against the governor limit for DML statements.

merge(masterRecord, duplicateIds, accessLevel)
Merges up to two records of the same sObject type into the master sObject record, deleting the others, and reparenting any related records.

Signature

public static List<Database.MergeResult> merge(SObject masterRecord, List<Id> duplicateIds, System.AccessLevel accessLevel)

Parameters

masterRecord
Type: SObject
The master sObject record the other records are merged into.

duplicateIds
Type: List<Id>
A list of IDs of up to two records to merge with the master. These records must be of the same sObject type as the master.

accessLevel
Type: System.AccessLevel
(Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

**Return Value**

Type: `List<Database.MergeResult>`

**Usage**

Each executed `merge` method counts against the governor limit for DML statements.

**merge (masterRecord, duplicateRecords, accessLevel)**

Merges up to two records of the same object type into the master sObject record, deleting the others, and reparenting any related records.

**Signature**

```java
public static List<Database.MergeResult> merge(SObject masterRecord, List<SObject> duplicateRecords, System.AccessLevel accessLevel)
```

**Parameters**

**masterRecord**

Type: `SObject`

The master sObject record the other sObjects are merged into.

**duplicateRecords**

Type: `List<SObject>`

A list of up to two sObject records to merge with the master. These sObjects must be of the same type as the master.

**accessLevel**

Type: `System.AccessLevel`

(Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

**Return Value**

Type: `List<Database.MergeResult>`

**Usage**

Each executed `merge` method counts against the governor limit for DML statements.
merge(masterRecord, duplicateId, allOrNone, accessLevel)

Merges the specified duplicate record into the master sObject record of the same type, optionally returning errors, if any, deleting the duplicate, and reparenting any related records. Merges only accounts, contacts, or leads.

Signature

public static Database.MergeResult merge(SObject masterRecord, Id duplicateId, Boolean allOrNone, System.AccessLevel accessLevel)

Parameters

masterRecord
Type: sObject
The master sObject record the duplicate record is merged into.

duplicateId
Type: ID
The ID of the record to merge with the master. This record must be of the same sObject type as the master.

allOrNone
Type: Boolean
Set to false to return any errors encountered in this operation as part of the returned result. If set to true, this method throws an exception if the operation fails. The default is true.

accessLevel
Type: System.AccessLevel
(Optional) The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value
Type: Database.MergeResult

Usage

If you use the accessLevel parameter to specify that the method runs in user mode, we report all encountered inaccessible fields. The way to retrieve the names of these inaccessible fields depends on the value of this method's allOrNone parameter, or the equivalent DmlOptions.optAllOrNone property. If you specify that:

- allOrNone=true or DmlOptions.optAllOrNone=true: Catch the DMLException and use the DMLException.getDMLFieldNames() method to retrieve the list of inaccessible fields. See Exception Class and Built-In Exceptions for more information.
- allOrNone=false or DmlOptions.optAllOrNone=false: For each failing record, we update the Database.Error object that results from the DML operation. Use the Error.getFields() method to retrieve the list of inaccessible fields. See the Error Class methods for more information.

Each executed merge method counts against the governor limit for DML statements.
**merge(masterRecord, duplicateRecord, allOrNone, accessLevel)**

Merges the specified duplicate sObject record into the master sObject of the same type, optionally returning errors, if any, deleting the duplicate, and reparenting any related records.

**Signature**

```java
public static Database.MergeResult merge(SObject masterRecord, SObject duplicateRecord, Boolean allOrNone, System.AccessLevel accessLevel)
```

**Parameters**

- **masterRecord**
  - Type: `sObject`
  - The master sObject record the duplicate record is merged into.

- **duplicateRecord**
  - Type: `sObject`
  - The sObject record to merge with the master. This sObject must be of the same type as the master.

- **allOrNone**
  - Type: `Boolean`
  - Set to `false` to return any errors encountered in this operation as part of the returned result. If set to `true`, this method throws an exception if the operation fails. The default is `true`.

- **accessLevel**
  - Type: `System.AccessLevel`
  - (Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

**Return Value**

- Type: `Database.MergeResult`

**Usage**

If you use the `accessLevel` parameter to specify that the method runs in user mode, we report all encountered inaccessible fields. The way to retrieve the names of these inaccessible fields depends on the value of this method's `allOrNone` parameter, or the equivalent `DmlOptions.optAllOrNone` property. If you specify that:

- `allOrNone=true` or `DmlOptions.optAllOrNone=true`: Catch the `DMLException` and use the `DMLException.getDMLFieldNames()` method to retrieve the list of inaccessible fields. See Exception Class and Built-In Exceptions for more information.

- `allOrNone=false` or `DmlOptions.optAllOrNone=false`: For each failing record, we update the `Database.Error` object that results from the DML operation. Use the `Error.getFields()` method to retrieve the list of inaccessible fields. See the Error Class methods for more information.

Each executed `merge` method counts against the governor limit for DML statements.
merge(masterRecord, duplicateIds, allOrNone, accessLevel)

Merges up to two records of the same sObject type into the master sObject record, optionally returning errors, if any, deleting the duplicates, and reparenting any related records.

**Signature**

```
public static List<Database.MergeResult> merge(SObject masterRecord, List<Id> duplicateIds, Boolean allOrNone, System.AccessLevel accessLevel)
```

**Parameters**

- `masterRecord`
  
  Type: `SObject`

  The master sObject record the other records are merged into.

- `duplicateIds`
  
  Type: `List<Id>`

  A list of IDs of up to two records to merge with the master. These records must be of the same sObject type as the master.

- `allOrNone`
  
  Type: `Boolean`

  Set to `false` to return any errors encountered in this operation as part of the returned result. If set to `true`, this method throws an exception if the operation fails. The default is `true`.

- `accessLevel`
  
  Type: `System.AccessLevel`

  (Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

**Return Value**

Type: `List<Database.MergeResult>`

**Usage**

If you use the `accessLevel` parameter to specify that the method runs in user mode, we report all encountered inaccessible fields. The way to retrieve the names of these inaccessible fields depends on the value of this method's `allOrNone` parameter, or the equivalent `DmlOptions.optAllOrNone` property. If you specify that:

- `allOrNone=true` or `DmlOptions.optAllOrNone=true`: Catch the `DMLException` and use the `DMLException.getDMLFieldNames()` method to retrieve the list of inaccessible fields. See Exception Class and Built-In Exceptions for more information.
- `allOrNone=false` or `DmlOptions.optAllOrNone=false`: For each failing record, we update the `Database.Error` object that results from the DML operation. Use the `Error.getFields()` method to retrieve the list of inaccessible fields. See the Error Class methods for more information.

Each executed `merge` method counts against the governor limit for DML statements.
merge (masterRecord, duplicateRecords, allOrNone, accessLevel)

Merges up to two records of the same object type into the master sObject record, optionally returning errors, if any, deleting the duplicates, and reparenting any related records.

Signature

public static List<Database.MergeResult> merge(SObject masterRecord, List<SObject> duplicateRecords, Boolean allOrNone, System.AccessLevel accessLevel)

Parameters

masterRecord
Type: sObject
The master sObject record the other sObjects are merged into.

duplicateRecords
Type: List<SObject>
A list of up to two sObject records to merge with the master. These sObjects must be of the same type as the master.

allOrNone
Type: Boolean
Set to false to return any errors encountered in this operation as part of the returned result. If set to true, this method throws an exception if the operation fails. The default is true.

accessLevel
Type: System.AccessLevel
(Optional) The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value

Type: List<Database.MergeResult>

Usage

If you use the accessLevel parameter to specify that the method runs in user mode, we report all encountered inaccessible fields. The way to retrieve the names of these inaccessible fields depends on the value of this method's allOrNone parameter, or the equivalent DmlOptions.optAllOrNone property. If you specify that:

- allOrNone=true or DmlOptions.optAllOrNone=true: Catch the DMLException and use the DMLException.getDMLFieldNames() method to retrieve the list of inaccessible fields. See Exception Class and Built-In Exceptions for more information.
- allOrNone=false or DmlOptions.optAllOrNone=false: For each failing record, we update the Database.Error object that results from the DML operation. Use the Error.getFields() method to retrieve the list of inaccessible fields. See the Error Class methods for more information.

Each executed merge method counts against the governor limit for DML statements.
**query (queryString)**

Creates a dynamic SOQL query at runtime.

**Signature**

```java
public static List<SObject> query(String queryString)
```

**Parameters**

- `queryString`
  - Type: `String`

**Return Value**

Type: `List on page 3177 <sObject>`

**Usage**

This method can be used wherever a static SOQL query can be used, such as in regular assignment statements and `for` loops. Unlike inline SOQL, fields in bind variables are not supported.

For more information, see Dynamic SOQL.

Each executed `query` method counts against the governor limit for SOQL queries.

**query (queryString, accessLevel)**

Creates a dynamic SOQL query at runtime.

**Signature**

```java
public static List<SObject> query(String queryString, System.AccessLevel accessLevel)
```

**Parameters**

- `queryString`
  - Type: `String`
- `accessLevel`
  - Type: `System.AccessLevel`

(Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

**Return Value**

Type: `List on page 3177 <sObject>`
Usage

This method can be used wherever a static SOQL query can be used, such as in regular assignment statements and for loops. Unlike inline SOQL, fields in bind variables are not supported.

For more information, see Dynamic SOQL.

Each executed query method counts against the governor limit for SOQL queries.

**queryWithBinds**(queryString, bindMap, accessLevel)

Creates a dynamic SOQL query at runtime. Bind variables in the query are resolved from the bindMap Map parameter directly with the key, rather than from Apex code variables.

**Signature**

public static List<SObject> queryWithBinds(String queryString, Map<String, Object> bindMap, System.AccessLevel accessLevel)

**Parameters**

**queryString**

Type: String

SOQL query that includes Apex bind variables or expressions preceded by a colon. All bind variables must have a key in the bindMap Map.

**bindMap**

Type: Map<String, Object>

A map that contains keys for each bind variable specified in the SOQL queryString and its value. The keys can’t be null or duplicates, and the values can’t be null or empty strings.

**accessLevel**

Type: System.AccessLevel

The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced.

**Return Value**

Type: List on page 3177<SObject>

Usage

This method can be used wherever a static SOQL query can be used, such as in regular assignment statements and for loops.

For more information, see Dynamic SOQL.

Each executed queryWithBinds method counts against the governor limit for SOQL queries.
Example

In this example, the SOQL query uses a bind variable for an Account name. Its value (Acme Inc.) is passed in to the method using the `nameBind` Map. The `accountName` variable isn’t (and doesn’t have to be) in scope when the query is executed within the method.

```java
public static List<Account> simpleBindingSoqlQuery(Map<String, Object> bindParams) {
    String queryString =
        'SELECT Id, Name ' +
        'FROM Account ' +
        'WHERE name = :name';
    return Database.queryWithBinds(
        queryString,
        bindParams,
        AccessLevel.USER_MODE
    );
}

String accountName = 'Acme Inc.';
Map<String, Object> nameBind = new Map<String, Object>{'name' => accountName};
List<Account> accounts = simpleBindingSoqlQuery(nameBind);
System.debug(accounts);
```

**rollback (databaseSavepoint)**

Restores the database to the state specified by the savepoint variable. Any emails submitted since the last savepoint are also rolled back and not sent.

**Signature**

```java
public static Void rollback(System.Savepoint databaseSavepoint)
```

**Parameters**

- `databaseSavepoint`
  - Type: System.Savepoint

**Return Value**

Type: Void

**Usage**

Note the following:

- Static variables are not reverted during a rollback. If you try to run the trigger again, the static variables retain the values from the first run.
- Each rollback counts against the governor limit for DML statements. You will receive a runtime error if you try to rollback the database additional times.
- The ID on an sObject inserted after setting a savepoint is not cleared after a rollback. Create an sObject to insert after a rollback. Attempting to insert the sObject using the variable created before the rollback fails because the sObject variable has an ID. Updating or upserting the sObject using the same variable also fails because the sObject is not in the database and, thus, cannot be updated.
For an example, see Transaction Control.

**setSavepoint()**

Returns a savepoint variable that can be stored as a local variable, then used with the rollback method to restore the database to that point.

**Signature**

```java
public static System.Savepoint setSavepoint()
```

**Return Value**

Type: System.Savepoint

**Usage**

Note the following:

- If you set more than one savepoint, then roll back to a savepoint that is not the last savepoint you generated, the later savepoint variables become invalid. For example, if you generated savepoint SP1 first, savepoint SP2 after that, and then you rolled back to SP1, the variable SP2 would no longer be valid. You will receive a runtime error if you try to use it.
- References to savepoints cannot cross trigger invocations because each trigger invocation is a new trigger context. If you declare a savepoint as a static variable then try to use it across trigger contexts, you will receive a run-time error.
- Each savepoint you set counts against the governor limit for DML statements.

For an example, see Transaction Control.

**undelete(recordToUndelete, allOrNone)**

Restores an existing sObject record, such as an individual account or contact, from your organization’s Recycle Bin.

**Signature**

```java
public static Database.UndeleteResult undelete(sObject recordToUndelete, Boolean allOrNone)
```

**Parameters**

- `recordToUndelete`
  Type: sObject
- `allOrNone`
  Type: Boolean

The optional `allOrNone` parameter specifies whether the operation allows partial success. If you specify `false` for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set `true`, an exception is thrown if the method is not successful.

**Return Value**

Type: Database.UndeleteResult
Usage

`undelete` is analogous to the UNDELETE statement in SQL.
Each executed `undelete` method counts against the governor limit for DML statements.

`undelete(recordsToUndelete, allOrNone)`
Restores one or more existing sObject records, such as individual accounts or contacts, from your organization’s Recycle Bin.

Signature

```java
public static Database.UndeleteResult[] undelete(sObject[] recordsToUndelete, Boolean allOrNone)
```

Parameters

- `recordsToUndelete`  
  Type: `sObject[]`
- `allOrNone`  
  Type: `Boolean`
  The optional `allOrNone` parameter specifies whether the operation allows partial success. If you specify `false` for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set `true`, an exception is thrown if the method is not successful.

Return Value

Type: `Database.UndeleteResult[]`

Usage

`undelete` is analogous to the UNDELETE statement in SQL.
Each executed `undelete` method counts against the governor limit for DML statements.

Example

The following example restores all accounts named ‘Universal Containers’. The `ALL ROWS` keyword queries all rows for both top-level and aggregate relationships, including deleted records and archived activities.

```java
Account[] savedAccts = [SELECT Id, Name FROM Account WHERE Name = 'Universal Containers' ALL ROWS];
Database.UndeleteResult[] UDR_Dels = Database.undelete(savedAccts);
```

`undelete(recordID, allOrNone)`
Restores an existing sObject record, such as an individual account or contact, from your organization’s Recycle Bin.
Signature

public static Database.UndeleteResult undelete(ID recordID, Boolean allOrNone)

Parameters

recordID
Type: ID

allOrNone
Type: Boolean

The optional allOrNone parameter specifies whether the operation allows partial success. If you specify false for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set true, an exception is thrown if the method is not successful.

Return Value

Type: Database.UndeleteResult

Usage

undelete is analogous to the UNDELETE statement in SQL.

Each executed undelete method counts against the governor limit for DML statements.

undelete(recordIDs, allOrNone)

Restores one or more existing sObject records, such as individual accounts or contacts, from your organization’s Recycle Bin.

Signature

public static Database.UndeleteResult[] undelete(ID[] recordIDs, Boolean allOrNone)

Parameters

RecordIDs
Type: ID[]

allOrNone
Type: Boolean

The optional allOrNone parameter specifies whether the operation allows partial success. If you specify false for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set true, an exception is thrown if the method is not successful.

Return Value

Type: Database.UndeleteResult[]
Usage

undelete is analogous to the UNDELETE statement in SQL.

Each executed undelete method counts against the governor limit for DML statements.

undelete(recordToUndelete, allOrNone, accessLevel)

Restores an existing sObject record, such as an individual account or contact, from your organization’s Recycle Bin.

Signature

public static Database.UndeleteResult undelete(SObject recordToUndelete, Boolean allOrNone, System.AccessLevel accessLevel)

Parameters

recordToUndelete
  Type: SObject

allOrNone
  Type: Boolean
  The optional allOrNone parameter specifies whether the operation allows partial success. If you specify false for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set true, an exception is thrown if the method is not successful.

accessLevel
  Type: System.AccessLevel
  (Optional) The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value

Type: Database.UndeleteResult

Usage

undelete is analogous to the UNDELETE statement in SQL.

Each executed undelete method counts against the governor limit for DML statements.

undelete(recordsToUndelete, allOrNone, accessLevel)

Restores one or more existing sObject records, such as individual accounts or contacts, from your organization’s Recycle Bin.

Signature

public static List<Database.UndeleteResult> undelete(List<SObject> recordsToUndelete, Boolean allOrNone, System.AccessLevel accessLevel)
Parameters

recordsToUndelete
  Type: List<sObject>

allOrNone
  Type: Boolean

The optional allOrNone parameter specifies whether the operation allows partial success. If you specify false for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set true, an exception is thrown if the method is not successful.

accessLevel
  Type: System.AccessLevel

(Optional) The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value

Type: List<Database.UndeleteResult>

Usage

undelete is analogous to the UNDELETE statement in SQL.

Each executed undelete method counts against the governor limit for DML statements.

undelete(recordID, allOrNone, accessLevel)

Restores an existing sObject record, such as an individual account or contact, from your organization's Recycle Bin.

Signature

public static Database.UndeleteResult undelete(Id recordID, Boolean allOrNone, System.AccessLevel accessLevel)

Parameters

recordID
  Type: Id

allOrNone
  Type: Boolean

The optional allOrNone parameter specifies whether the operation allows partial success. If you specify false for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set true, an exception is thrown if the method is not successful.

accessLevel
  Type: System.AccessLevel
(Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

**Return Value**
Type: `Database.UndeleteResult`

**Usage**

`undelete` is analogous to the UNDELETE statement in SQL.

Each executed `undelete` method counts against the governor limit for DML statements.

`undelete(recordIDs, allOrNone, accessLevel)`
Restores one or more existing sObject records, such as individual accounts or contacts, from your organization’s Recycle Bin.

**Signature**

```java
public static List<Database.UndeleteResult> undelete(List<Id> recordIDs, Boolean allOrNone, System.AccessLevel accessLevel)
```

**Parameters**

- `recordIDs`
  Type: `List<Id>`

- `allOrNone`
  Type: `Boolean`
  The optional `allOrNone` parameter specifies whether the operation allows partial success. If you specify `false` for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set `true`, an exception is thrown if the method is not successful.

- `accessLevel`
  Type: `System.AccessLevel`
  (Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

**Return Value**
Type: `List<Database.UndeleteResult>`

**Usage**

`undelete` is analogous to the UNDELETE statement in SQL.

Each executed `undelete` method counts against the governor limit for DML statements.
**update**(recordToUpdate, allOrNone)

Modifies an existing sObject record, such as an individual account or contact, in your organization’s data.

**Signature**

```
public static Database.SaveResult update(sObject recordToUpdate, Boolean allOrNone)
```

**Parameters**

- **recordToUpdate**
  - Type: sObject
- **allOrNone**
  - Type: Boolean

The optional `allOrNone` parameter specifies whether the operation allows partial success. If you specify `false` for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set `true`, an exception is thrown if the method is not successful.

**Return Value**

Type: `Database.SaveResult`

**Usage**

`update` is analogous to the UPDATE statement in SQL.

Apex classes and triggers saved (compiled) using API version 15.0 and higher produce a runtime error if you assign a String value that is too long for the field.

Each executed `update` method counts against the governor limit for DML statements.

**Example**

The following example updates the `BillingCity` field on a single account.

```java
Account a = new Account(Name='SFDC');
insert(a);

Account myAcct =
    [SELECT Id, Name, BillingCity
     FROM Account WHERE Id = :a.Id];
myAcct.BillingCity = 'San Francisco';

Database.SaveResult SR =
    Database.update(myAcct);
```

**update**(recordsToUpdate, allOrNone)

Modifies one or more existing sObject records, such as individual accounts or contacts, in your organization’s data.
Signature

public static Database.SaveResult[] update(sObject[] recordsToUpdate, Boolean allOrNone)

Parameters

recordsToUpdate
Type: sObject []

allOrNone
Type: Boolean

The optional allOrNone parameter specifies whether the operation allows partial success. If you specify false for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set true, an exception is thrown if the method is not successful.

Return Value

Type: Database.SaveResult[]

Usage

update is analogous to the UPDATE statement in SQL.

Each executed update method counts against the governor limit for DML statements.

update(recordToUpdate, dmlOptions)

Modifies an existing sObject record, such as an individual account or contact, in your organization's data.

Signature

public static Database.SaveResult update(sObject recordToUpdate, Database.DmlOptions dmlOptions)

Parameters

recordToUpdate
Type: sObject

dmlOptions
Type: Database.DmlOptions

The optional dmlOptions parameter specifies additional data for the transaction, such as assignment rule information or rollback behavior when errors occur during record insertions.

Return Value

Type: Database.SaveResult

Usage

update is analogous to the UPDATE statement in SQL.
Apex classes and triggers saved (compiled) using API version 15.0 and higher produce a runtime error if you assign a String value that is too long for the field.

Each executed `update` method counts against the governor limit for DML statements.

### `update(recordsToUpdate, dmlOptions)`

Modifies one or more existing sObject records, such as individual accounts or contacts, in your organization's data.

**Signature**

```java
public static Database.SaveResult[] update(sObject[] recordsToUpdate, Database.DMLOptions dmlOptions)
```

**Parameters**

- `recordsToUpdate`  
  Type: `sObject []`

- `dmlOptions`  
  Type: `Database.DMLOptions`

  The optional `dmlOptions` parameter specifies additional data for the transaction, such as assignment rule information or rollback behavior when errors occur during record insertions.

**Return Value**

Type: `Database.SaveResult[]`

**Usage**

`update` is analogous to the UPDATE statement in SQL.

Apex classes and triggers saved (compiled) using API version 15.0 and higher produce a runtime error if you assign a String value that is too long for the field.

Each executed `update` method counts against the governor limit for DML statements.

### `update(recordToUpdate, allOrNone, accessLevel)`

Modifies an existing sObject record, such as an individual account or contact, in your organization's data.

**Signature**

```java
public static Database.SaveResult update(SObject recordToUpdate, Boolean allOrNone, System.AccessLevel accessLevel)
```

**Parameters**

- `recordToUpdate`  
  Type: `SObject`

- `allOrNone`  
  Type: `Boolean`
The optional allOrNone parameter specifies whether the operation allows partial success. If you specify false for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set true, an exception is thrown if the method is not successful.

accessLevel
Type: System.AccessLevel
(Optional) The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value
Type: Database.SaveResult

Usage
If you use the accessLevel parameter to specify that the method runs in user mode, we report all encountered inaccessible fields. The way to retrieve the names of these inaccessible fields depends on the value of this method’s allOrNone parameter, or the equivalent DmlOptions.optAllOrNone property. If you specify that:

- allOrNone=true or DmlOptions.optAllOrNone=true: Catch the DMLException and use the DMLException.getDMLFieldNames() method to retrieve the list of inaccessible fields. See Exception Class and Built-In Exceptions for more information.
- allOrNone=false or DmlOptions.optAllOrNone=false: For each failing record, we update the Database.Error object that results from the DML operation. Use the Error.getFields() method to retrieve the list of inaccessible fields. See the Error Class methods for more information.

update(recordsToUpdate, allOrNone, accessLevel)
Modifies one or more existing sObject records, such as individual accounts or contacts, in your organization’s data.

Signature
public static List<Database.SaveResult> update(List<SObject> recordsToUpdate, Boolean allOrNone, System.AccessLevel accessLevel)

Parameters
recordsToUpdate
Type: List<SObject>

allOrNone
Type: Boolean
The optional allOrNone parameter specifies whether the operation allows partial success. If you specify false for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set true, an exception is thrown if the method is not successful.

accessLevel
Type: System.AccessLevel
(Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

**Return Value**
Type: `List<Database.SaveResult>`

**Usage**
If you use the `accessLevel` parameter to specify that the method runs in user mode, we report all encountered inaccessible fields. The way to retrieve the names of these inaccessible fields depends on the value of this method's `allOrNone` parameter, or the equivalent `DmlOptions.optAllOrNone` property. If you specify that:

- `allOrNone=true` or `DmlOptions.optAllOrNone=true`: Catch the `DMLException` and use the `DMLException.getDMLFieldNames()` method to retrieve the list of inaccessible fields. See Exception Class and Built-In Exceptions for more information.
- `allOrNone=false` or `DmlOptions.optAllOrNone=false`: For each failing record, we update the `Database.Error` object that results from the DML operation. Use the `Error.getFields()` method to retrieve the list of inaccessible fields. See the Error Class methods for more information.

**`update(recordToUpdate, dmlOptions, accessLevel)`**
Modifies an existing sObject record, such as an individual account or contact, in your organization's data.

**Signature**
```java
public static Database.SaveResult update(SObject recordToUpdate, Database.DMLOptions dmlOptions, System.AccessLevel accessLevel)
```

**Parameters**
- `recordToUpdate`
  Type: `SObject`
- `dmlOptions`
  Type: `Database.DMLOptions`
  The optional `dmlOptions` parameter specifies additional data for the transaction, such as assignment rule information or rollback behavior when errors occur during record insertions.
- `accessLevel`
  Type: `System.AccessLevel`
  (Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

**Return Value**
Type: `Database.SaveResult`
update(recordsToUpdate, dmlOptions, accessLevel)

Modifies one or more existing sObject records, such as individual accounts or contacts, in your organization’s data.

Signature

public static List<Database.SaveResult> update(List<SObject> recordsToUpdate, Database.DMLOptions dmlOptions, System.AccessLevel accessLevel)

Parameters

recordsToUpdate
- Type: List<sObject>

dmlOptions
- Type: Database.DMLOptions

The optional dmlOptions parameter specifies additional data for the transaction, such as assignment rule information or rollback behavior when errors occur during record insertions.

accessLevel
- Type: System.AccessLevel

(Optional) The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value

Type: List<Database.SaveResult>

upsert(recordToUpsert, externalIdField, allOrNone)

Creates a new sObject record or updates an existing sObject record within a single statement, using a specified field to determine the presence of existing objects, or the ID field if no field is specified.

Signature

public static Database.UpsertResult upsert(sObject recordToUpsert, Schema.SObjectField externalIdField, Boolean allOrNone)

Parameters

recordToUpsert
- Type: sObject

externalIdField
- Type: Schema.SObjectField

(Optional) The externalIdField parameter is of type Schema.SObjectField, that is, a field token. Find the token for the field by using the fields special method. For example, Schema.SObjectField f = Account.Fields.MyExternalId. The externalIdField parameter is the field that upsert() uses to match sObjects with existing records. This field can be a custom field marked as external ID, or a standard field with the idLookup attribute.
Note: If `externalIdField` isn’t specified, then the ID field is used to determine a match with existing records.

**allOrNone**
Type: `Boolean`

The optional `allOrNone` parameter specifies whether the operation allows partial success. If you specify `false` for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set `true`, an exception is thrown if the method is not successful.

**Return Value**
Type: `Database.UpsertResult`

**Usage**
Apex classes and triggers saved (compiled) using API version 15.0 and higher produce a runtime error if you assign a String value that is too long for the field.

Each executed `upsert` method counts against the governor limit for DML statements.

For more information on how the `upsert` operation works, see the `upsert()` statement.

**upsert**(recordsToUpsert, externalIdField, allOrNone)**

Creates new sObject records or updates existing sObject records within a single statement, using a specified field to determine the presence of existing objects, or the ID field if no field is specified.

**Signature**
```
public static Database.UpsertResult[] upsert(sObject[] recordsToUpsert, Schema.SObjectField externalIdField, Boolean allOrNone)
```

**Parameters**
- **recordsToUpsert**
  Type: `sObject` []

- **externalIdField**
  Type: `Schema.SObjectField`

  (Optional) The `externalIdField` is of type `Schema.SObjectField`, that is, a field token. Find the token for the field by using the `fields` special method. For example, `Schema.SObjectField f = Account.Fields.MyExternalId`. The `externalIdField` parameter is the field that `upsert()` uses to match sObjects with existing records. This field can be a custom field marked as external ID, or a standard field with the `idLookup` attribute.

  **Note:** If `externalIdField` isn’t specified, then the ID field is used to determine a match with existing records.

- **allOrNone**
  Type: `Boolean`

  The optional `allOrNone` parameter specifies whether the operation allows partial success. If you specify `false` for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter isn’t set or is set `true`, an exception is thrown if the method
isn’t successful. If the parameter is specified as false and a before-trigger assigns an invalid value to a field, the partial set of valid records isn’t inserted.

Return Value
Type: Database.UpsertResult[]

Usage
Apex classes and triggers saved (compiled) using API version 15.0 and higher produce a runtime error if you assign a String value that is too long for the field.
Each executed upsert method counts against the governor limit for DML statements.
For more information on how the upsert operation works, see the upsert() statement.

upsert(recordToUpsert, externalIdField, allOrNone, accessLevel)

Creates a new sObject record or updates an existing sObject record within a single statement, using a specified field to determine the presence of existing objects, or the ID field if no field is specified.

Signature
public static Database.UpsertResult upsert(SObject recordToUpsert, Schema.SObjectField externalIdField, Boolean allOrNone, System.AccessLevel accessLevel)

Parameters
recordToUpsert
Type: SObject

externalIdField
Type: Schema.SObjectField
(Optional) The externalIdField is of type Schema.SObjectField, that is, a field token. Find the token for the field by using the fields special method. For example, Schema.SObjectField f = Account.Fields.MyExternalId. The externalIdField parameter is the field that upsert() uses to match sObjects with existing records. This field can be a custom field marked as external ID, or a standard field with the idLookup attribute.

Note: If externalIdField isn’t specified, then the ID field is used to determine a match with existing records.

allOrNone
Type: Boolean
The optional allOrNone parameter specifies whether the operation allows partial success. If you specify false for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter is not set or is set true, an exception is thrown if the method is not successful.

accessLevel
Type: System.AccessLevel
(Optional) The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are
ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value
Type: Database.UpsertResult

Usage
If you use the accessLevel parameter to specify that the method runs in user mode, we report all encountered inaccessible fields. The way to retrieve the names of these inaccessible fields depends on the value of this method’s allOrNone parameter, or the equivalent DmlOptions.optAllOrNone property. If you specify that:

- allOrNone=true or DmlOptions.optAllOrNone=true: Catch the DMLException and use the DMLException.getDMLFieldNames() method to retrieve the list of inaccessible fields. See Exception Class and Built-In Exceptions for more information.
- allOrNone=false or DmlOptions.optAllOrNone=false: For each failing record, we update the Database.Error object that results from the DML operation. Use the Error.getFields() method to retrieve the list of inaccessible fields. See the Error Class methods for more information.

Apex classes and triggers saved (compiled) using API version 15.0 and higher produce a runtime error if you assign a String value that is too long for the field.

Each executed upsert method counts against the governor limit for DML statements.

For more information on how the upsert operation works, see the upsert() statement.

upsert(recordsToUpsert, externalIdField, allOrNone, accessLevel)
Creates new sObject records or updates existing sObject records within a single statement, using a specified field to determine the presence of existing objects, or the ID field if no field is specified.

Signature
public static List<Database.UpsertResult> upsert(List<SObject> recordsToUpsert, Schema.SObjectField externalIdField, Boolean allOrNone, System.AccessLevel accessLevel)

Parameters
recordsToUpsert
Type: List<SObject>

externalIdField
Type: Schema.SObjectField
(Optional) The externalIdField is of type Schema.SObjectField, that is, a field token. Find the token for the field by using the fields special method. For example, Schema.SObjectField f = Account.Fields.MyExternalId. The externalIdField parameter is the field that upsert () uses to match sObjects with existing records. This field can be a custom field marked as external ID, or a standard field with the idLookup attribute.

Note: If externalIdField isn’t specified, then the ID field is used to determine a match with existing records.

allOrNone
Type: Boolean
The optional `allOrNone` parameter specifies whether the operation allows partial success. If you specify `false` for this parameter and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why. If the parameter isn’t set or is set `true`, an exception is thrown if the method isn’t successful. If the parameter is specified as `false` and a before-trigger assigns an invalid value to a field, the partial set of valid records isn’t inserted.

**accessLevel**

Type: `System.AccessLevel`

(Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

**Return Value**

Type: `List<Database.UpsertResult>`

**Usage**

If you use the `accessLevel` parameter to specify that the method runs in user mode, we report all encountered inaccessible fields. The way to retrieve the names of these inaccessible fields depends on the value of this method’s `allOrNone` parameter, or the equivalent `DmlOptions.optAllOrNone` property. If you specify that:

- `allOrNone=true` or `DmlOptions.optAllOrNone=true`: Catch the `DMLException` and use the `DMLException.getDMLFieldNames()` method to retrieve the list of inaccessible fields. See Exception Class and Built-In Exceptions for more information.
- `allOrNone=false` or `DmlOptions.optAllOrNone=false`: For each failing record, we update the `Database.Error` object that results from the DML operation. Use the `Error.getFields()` method to retrieve the list of inaccessible fields. See the Error Class methods for more information.

Apex classes and triggers saved (compiled) using API version 15.0 and higher produce a runtime error if you assign a String value that is too long for the field.

Each executed `upsert` method counts against the governor limit for DML statements.

For more information on how the upsert operation works, see the `upsert()` statement.

**updateAsync(sobjects, callback)**

Initiates requests to update external object data on the relevant external systems. The requests are executed asynchronously, as background operations, and are sent to the external systems that are defined by the external objects’ associated external data sources. Allows referencing a callback class whose `processSave` method is called for each record after the remote operations are completed.

**Signature**

```
public static List<Database.SaveResult> updateAsync(List<SObject> sobjects,
DataSource.AsyncSaveCallback callback)
```

**Parameters**

- `sobjects`
  Type: `List<SObject>`

---

2997
List of external object records to modify.

callback
   Type: DataSource.AsyncSaveCallback

   The callback object that contains the state in the originating context and an action (the processSave method) that executes after the insert operation is completed. Use the action callback to update org data according to the operation’s results. The callback object must extend DataSource.AsyncSaveCallback.

Return Value
Type: List<Database.SaveResult>

Status results for the update operation. Each result corresponds to a record processed by this asynchronous operation and is associated with a unique identifier (asyncLocator). The asyncLocator value is included in the errors array of the result. You can retrieve this identifier with Database.getAsyncLocator(). Retrieve the final result with Database.getAsyncSaveResult().

updateAsync(sobject, callback)

Initiates a request to update external object data on the relevant external system. The request is executed asynchronously, as a background operation, and is sent to the external system that’s defined by the external object’s associated external data source. Allows referencing a callback class whose processSave method is called after the remote operation is completed.

Signature

public static Database.SaveResult updateAsync(SObject sobject, DataSource.AsyncSaveCallback callback)

Parameters

sobject
   Type: SObject

   External object record to modify.

callback
   Type: DataSource.AsyncSaveCallback

   The callback object that contains the state in the originating context and an action (the processSave method) that executes after the insert operation is completed. Use the action callback to update org data according to the operation’s results. The callback object must extend DataSource.AsyncSaveCallback.

Return Value
Type: Database.SaveResult

Status result for the insert operation. The result corresponds to a record processed by this asynchronous operation and is associated with a unique identifier (asyncLocator). The asyncLocator value is included in the errors array of the result. You can retrieve this identifier with Database.getAsyncLocator(). Retrieve the final result with Database.getAsyncSaveResult().

updateAsync(sobjects)

Initiates requests to update external object data on the relevant external systems. The requests are executed asynchronously, as background operations, and are sent to the external systems that are defined by the external objects’ associated external data sources.
Signature

```java
public static List<Database.SaveResult> updateAsync(List<SObject> sobjects)
```

Parameters

- **sobjects**  
  Type: List<SObject>  
  List of external object records to modify.

Return Value

Type: List<Database.SaveResult>

Status results for the update operation. Each result corresponds to a record processed by this asynchronous operation and is associated with a unique identifier (asyncLocator). The asyncLocator value is included in the errors array of the result. You can retrieve this identifier with `Database.getAsyncLocator()`. Retrieve the final result with `Database.getAsyncSaveResult()`.

**updateAsync(sobject)**

Initiates a request to update external object data on the relevant external system. The request is executed asynchronously, as a background operation, and is sent to the external system that’s defined by the external object’s associated external data source.

Signature

```java
public static Database.SaveResult updateAsync(SObject sobject)
```

Parameters

- **sobject**  
  Type: SObject  
  External object record to modify.

Return Value

Type: Database.SaveResult

Status result for the insert operation. The result corresponds to a record processed by this asynchronous operation and is associated with a unique identifier (asyncLocator). The asyncLocator value is included in the errors array of the result. You can retrieve this identifier with `Database.getAsyncLocator()`. Retrieve the final result with `Database.getAsyncSaveResult()`.

**updateAsync(sobjects, callback, accessLevel)**

Initiates requests to update external object data on the relevant external systems. The requests are executed asynchronously, as background operations, and are sent to the external systems that are defined by the external objects’ associated external data sources. Allows referencing a callback class whose `processSave` method is called for each record after the remote operations are completed.

Signature

```java
public static List<Database.SaveResult> updateAsync(List<SObject> sobjects, DataSource.AsyncSaveCallback callback, System.AccessLevel accessLevel)
```
Parameters

`sobjects`
  Type: List<SObject>
  List of external object records to modify.

`callback`
  Type: DataSource.AsyncSaveCallback
  The callback object that contains the state in the originating context and an action (the `processSave` method) that executes after the insert operation is completed. The execution is in system mode regardless of the `accessLevel` parameter. Use the action callback to update org data according to the operation's results. The callback object must extend `DataSource.AsyncSaveCallback`.

`accessLevel`
  Type: System.AccessLevel
  (Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value

Type: List<Database.SaveResult>
Status results for the update operation. Each result corresponds to a record processed by this asynchronous operation and is associated with a unique identifier (`asyncLocator`). The `asyncLocator` value is included in the errors array of the result. You can retrieve this identifier with `Database.getAsycnLocator()`. Retrieve the final result with `Database.getAsyncSaveResult()`.

`updateAsync(sobject, callback, accessLevel)`
Initiates a request to update external object data on the relevant external system. The request is executed asynchronously, as a background operation, and is sent to the external system that's defined by the external object's associated external data source. Allows referencing a callback class whose `processSave` method is called after the remote operation is completed.

Signature

`public static Database.SaveResult updateAsync(SObject sobject, DataSource.AsyncSaveCallback callback, System.AccessLevel accessLevel)`

Parameters

`sobject`
  Type: SObject
  External object record to modify.

`callback`
  Type: DataSource.AsyncSaveCallback
  The callback object that contains the state in the originating context and an action (the `processSave` method) that executes after the insert operation is completed. The execution is in system mode regardless of the `accessLevel` parameter. Use the action callback to update org data according to the operation's results. The callback object must extend `DataSource.AsyncSaveCallback`. 
accessLevel
Type: System.AccessLevel
(Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value
Type: Database.SaveResult
Status result for the insert operation. The result corresponds to a record processed by this asynchronous operation and is associated with a unique identifier (`asyncLocator`). The `asyncLocator` value is included in the errors array of the result. You can retrieve this identifier with `Database.getAsyncLocator()`. Retrieve the final result with `Database.getAsyncSaveResult()`.

updateAsync(sobjects, accessLevel)
Initiates requests to update external object data on the relevant external systems. The requests are executed asynchronously, as background operations, and are sent to the external systems that are defined by the external objects' associated external data sources.

Signature
```
public static List<Database.SaveResult> updateAsync(List<SObject> sobjects,
System.AccessLevel accessLevel)
```

Parameters
sobjects
Type: List<SObject>
List of external object records to modify.

accessLevel
Type: System.AccessLevel
(Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value
Type: List<Database.SaveResult>
Status results for the update operation. Each result corresponds to a record processed by this asynchronous operation and is associated with a unique identifier (`asyncLocator`). The `asyncLocator` value is included in the errors array of the result. You can retrieve this identifier with `Database.getAsyncLocator()`. Retrieve the final result with `Database.getAsyncSaveResult()`.

updateAsync(sobject, accessLevel)
Initiates a request to update external object data on the relevant external system. The request is executed asynchronously, as a background operation, and is sent to the external system that’s defined by the external object’s associated external data source.
Signature

`public static Database.SaveResult updateAsync(SObject sobject, System.AccessLevel accessLevel)`

Parameters

- `sobject`
  Type: `SObject`
  External object record to modify.
- `accessLevel`
  Type: `System.AccessLevel`
  (Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value

Type: `Database.SaveResult`

Status result for the insert operation. The result corresponds to a record processed by this asynchronous operation and is associated with a unique identifier (`asyncLocator`). The `asyncLocator` value is included in the errors array of the result. You can retrieve this identifier with `Database.getAsyncLocator()`. Retrieve the final result with `Database.getAsyncSaveResult()`.

`updateImmediate(sobjects)`

Initiates requests to update external object data on the relevant external systems. The requests are executed synchronously and are sent to the external systems that are defined by the external objects’ associated external data sources. If the Apex transaction contains pending changes, the synchronous operations can’t be completed and throw exceptions.

Signature

`public static List<Database.SaveResult> updateImmediate(List<SObject> sobjects)`

Parameters

- `sobjects`
  Type: `List<SObject>`
  List of external object records to modify.

Return Value

Type: `List<Database.SaveResult>`

Status results for the update operation.
Usage

The operation allows partial success. If one or more record updates fail, the method doesn’t throw an exception and the remainder of the DML operation can still succeed. The returned `SaveResult` objects indicate whether the operation was successful. If it wasn’t successful, the objects also return the error code and description.

**updateImmediate(sobject)**

Initiates a request to update external object data on the relevant external system. The request is executed synchronously and is sent to the external system that’s defined by the external object’s associated external data source. If the Apex transaction contains pending changes, the synchronous operation can’t be completed and throws an exception.

**Signature**

```java
public static Database.SaveResult updateImmediate(SObject sobject)
```

**Parameters**

- `sobject`  
  Type: `SObject`  
  External object record to modify.

**Return Value**

Type: `Database.SaveResult`  
Status result for the update operation.

**Usage**

If a record update fails, the method doesn’t throw an exception. The returned `SaveResult` object indicates whether the operation was successful. If it wasn’t successful, the object returns the error code and description.

**updateImmediate(sobjects, accessLevel)**

Initiates requests to update external object data on the relevant external systems. The requests are executed synchronously and are sent to the external systems that are defined by the external objects’ associated external data sources. If the Apex transaction contains pending changes, the synchronous operations can’t be completed and throw exceptions.

**Signature**

```java
public static List<Database.SaveResult> updateImmediate(List<SObject> sobjects, System.AccessLevel accessLevel)
```

**Parameters**

- `sobjects`  
  Type: `List<SObject>`  
  List of external object records to modify.

- `accessLevel`  
  Type: `System.AccessLevel`
(Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value
Type: `List<Database.SaveResult>`
Status results for the update operation.

Usage
The operation allows partial success. If one or more record updates fail, the method doesn’t throw an exception and the remainder of the DML operation can still succeed. The returned `SaveResult` objects indicate whether the operation was successful. If it wasn’t successful, the objects also return the error code and description.

`updateImmediate(sobject, accessLevel)`
Initiates a request to update external object data on the relevant external system. The request is executed synchronously and is sent to the external system that’s defined by the external object’s associated external data source. If the Apex transaction contains pending changes, the synchronous operation can’t be completed and throws an exception.

Signature
```java
public static Database.SaveResult updateImmediate(SObject sobject, System.AccessLevel accessLevel)
```

Parameters
- `sobject`  
  Type: `SObject`
  External object record to modify.
- `accessLevel`  
  Type: `System.AccessLevel`
  (Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value
Type: `Database.SaveResult`
Status result for the update operation.

Usage
If a record update fails, the method doesn’t throw an exception. The returned `SaveResult` object indicates whether the operation was successful. If it failed, the object returns the error code and description.
Date Class

Contains methods for the Date primitive data type.

Namespace

System

Usage

For more information on Dates, see Date Data Type.

Date Methods

The following are methods for `Date`.

IN THIS SECTION:

- `addDays(additionalDays)`
  Adds the specified number of additional days to a Date.
- `addMonths(additionalMonths)`
  Adds the specified number of additional months to a Date.
- `addYears(additionalYears)`
  Adds the specified number of additional years to a Date.
- `day()`
  Returns the day-of-month component of a Date.
- `dayOfYear()`
  Returns the day-of-year component of a Date.
- `daysBetween(secondDate)`
  Returns the number of days between the Date that called the method and the specified date.
- `daysInMonth(year, month)`
  Returns the number of days in the month for the specified `year` and `month` (1=Jan).
- `format()`
  Returns the Date as a string using the locale of the context user.
- `isLeapYear(year)`
  Returns `true` if the specified year is a leap year.
- `isSameDay(dateToCompare)`
  Returns `true` if the Date that called the method is the same as the specified date.
- `month()`
  Returns the month component of a Date (1=Jan).
- `monthsBetween(secondDate)`
  Returns the number of months between the Date that called the method and the specified date, ignoring the difference in days.
- `newInstance(year, month, day)`
  Constructs a Date from Integer representations of the `year`, `month` (1=Jan), and `day`.
parse(stringDate)
Constructs a Date from a String. The format of the String depends on the local date format.

today()
Returns the current date in the current user’s time zone.
toStartOfMonth()
Returns the first of the month for the Date that called the method.
toStartOfWeek()
Returns the start of the week for the Date that called the method, depending on the context user’s locale.
valueOf(stringDate)
Returns a Date that contains the value of the specified String.
valueOf(fieldValue)
Converts the specified object to a Date. Use this method to convert a history tracking field value or an object that represents a Date value.
year()
Returns the year component of a Date

addDays (additionalDays)
Adds the specified number of additional days to a Date.

Signature

```java
class public Date addDays(Integer additionalDays)
```

Parameters

- **additionalDays**
  - Type: Integer

Return Value

Type: Date

Example

```java
Date myDate = Date.newInstance(1960, 2, 17);
Date newDate = myDate.addDays(2);
```

addMonths (additionalMonths)
Adds the specified number of additional months to a Date

Signature

```java
class public Date addMonths(Integer additionalMonths)
```
Parameters

*additionalMonths*
  Type: Integer

Return Value

Type: Date

Example

date myDate = date.newInstance(1990, 11, 21);
date newDate = myDate.addMonths(3);
date expectedDate = date.newInstance(1991, 2, 21);
system.assertEquals(expectedDate, newDate);

**addYears (additionalYears)**

Adds the specified number of additional years to a Date

Signature

public Date addYears(Integer additionalYears)

Parameters

*additionalYears*
  Type: Integer

Return Value

Type: Date

Example

date myDate = date.newInstance(1983, 7, 15);
date newDate = myDate.addYears(2);
date expectedDate = date.newInstance(1985, 7, 15);
system.assertEquals(expectedDate, newDate);

day ()

Returns the day-of-month component of a Date.

Signature

public Integer day()
Example

date myDate = date.newInstance(1989, 4, 21);
Integer day = myDate.day();
System.assertEquals(21, day);

dayOfYear()
Returns the day-of-year component of a Date.

Signature

public Integer dayOfYear()

Return Value
Type: Integer

Example

date myDate = date.newInstance(1998, 10, 21);
Integer day = myDate.dayOfYear();
System.assertEquals(294, day);

daysBetween(secondDate)
Returns the number of days between the Date that called the method and the specified date.

Signature

public Integer daysBetween(Date secondDate)

Parameters

secondDate
Type: Date

Return Value
Type: Integer

Usage
If the Date that calls the method occurs after the secondDate, the return value is negative.

Example

Date startDate = Date.newInstance(2008, 1, 1);
Date dueDate = Date.newInstance(2008, 1, 30);
Integer numberDaysDue = startDate.daysBetween(dueDate);
**daysInMonth(year, month)**

Returns the number of days in the month for the specified `year` and `month` (1=Jan).

**Signature**

```java
public static Integer daysInMonth(Integer year, Integer month)
```

**Parameters**

- **year**
  - Type: `Integer`
- **month**
  - Type: `Integer`

**Return Value**

Type: `Integer`

**Example**

The following example finds the number of days in the month of February in the year 1960.

```java
Integer numberDays = date.daysInMonth(1960, 2);
```

**format()**

Returns the Date as a string using the locale of the context user

**Signature**

```java
public String format()
```

**Return Value**

Type: `String`

**Example**

```java
// In American-English locale
date myDate = date.newInstance(2001, 3, 21);
String dayString = myDate.format();
system.assertEquals('3/21/2001', dayString);
```

**isLeapYear(year)**

Returns `true` if the specified year is a leap year.

**Signature**

```java
public static Boolean isLeapYear(Integer year)
```
Parameters

*year*
Type: *Integer*

Return Value
Type: *Boolean*

Example

```java
system.assert(Date.isLeapYear(2004));
```

**isSameDay(dateToCompare)**

Returns *true* if the Date that called the method is the same as the specified date.

Signature

```java
public Boolean isSameDay(Date dateToCompare)
```

Parameters

*dateToCompare*
Type: *Date*

Return Value
Type: *Boolean*

Example

```java
date myDate = date.today();
date dueDate = date.newInstance(2008, 1, 30);
boolean dueNow = myDate.isSameDay(dueDate);
```

**month()**

Returns the month component of a Date (1=Jan).

Signature

```java
public Integer month()
```

Return Value
Type: *Integer*
Example

date myDate = date.newInstance(2004, 11, 21);
Integer month = myDate.month();
system.assertEquals(11, month);

monthsBetween(secondDate)
Returns the number of months between the Date that called the method and the specified date, ignoring the difference in days.

Signature

public Integer monthsBetween(Date secondDate)

Parameters

secondDate
Type: Date

Return Value

Type: Integer

Example

Date firstDate = Date.newInstance(2006, 12, 2);
Date secondDate = Date.newInstance(2012, 12, 8);
Integer monthsBetween = firstDate.monthsBetween(secondDate);
System.assertEquals(72, monthsBetween);

newInstance(year, month, day)
Constructs a Date from Integer representations of the year, month (1=Jan), and day.

Signature

public static Date newInstance(Integer year, Integer month, Integer day)

Parameters

year
Type: Integer

month
Type: Integer

day
Type: Integer

Return Value

Type: Date
Example
The following example creates the date February 17th, 1960:

```java
Date myDate = date.newInstance(1960, 2, 17);
```

**parse(stringDate)**
Constructs a Date from a String. The format of the String depends on the local date format.

**Signature**

```
public static Date parse(String stringDate)
```

**Parameters**

stringDate
Type: String

**Return Value**

Type: Date

**Example**
The following example works in some locales.

```java
date mydate = date.parse('12/27/2009');
```

**today()**
Returns the current date in the current user's time zone.

**Signature**

```
public static Date today()
```

**Return Value**

Type: Date

**toStartOfMonth()**
Returns the first of the month for the Date that called the method.

**Signature**

```
pUBLIC Date toStartOfMonth()
```

**Return Value**

Type: Date
**Example**

date myDate = date.newInstance(1987, 12, 17);
date firstDate = myDate.toStartOfMonth();
date expectedDate = date.newInstance(1987, 12, 1);
system.assertEquals(expectedDate, firstDate);

toStartOfWeek()

Returns the start of the week for the Date that called the method, depending on the context user's locale.

**Signature**

```
public Date toStartOfWeek()
```

**Return Value**

Type: Date

**Example**

For example, the start of a week is Sunday in the United States locale, and Monday in European locales. For example:

```
Date myDate = Date.today();
Date weekStart = myDate.toStartOfWeek();
```

valueOf(stringDate)

Returns a Date that contains the value of the specified String.

**Signature**

```
public static Date valueOf(String stringDate)
```

**Parameters**

```
stringDate
    Type: String
```

**Return Value**

Type: Date

**Usage**

The specified string should use the standard date format "yyyy-MM-dd HH:mm:ss" in the local time zone.

**Example**

```
string year = '2008';
string month = '10';
```
string day = '5';
string hour = '12';
string minute = '20';
string second = '20';
string stringDate = year + '-' + month + '-' + day + ' ' + hour + ':' + minute + ':' + second;
Date myDate = date.valueOf(stringDate);

valueOf(fieldValue)
Converts the specified object to a Date. Use this method to convert a history tracking field value or an object that represents a Date value.

Signature
public static Date valueOf(Object fieldValue)

Parameters
 fieldValue
  Type: Object

Return Value
 Type: Date

Usage
Use this method with the OldValue or NewValue fields of history sObjects, such as AccountHistory, when the field is a Date field.

Example
This example converts history tracking fields to Date values.

List<AccountHistory> ahlist = [SELECT Field,OldValue,NewValue FROM AccountHistory];
for(AccountHistory ah : ahlist) {
    System.debug('Field: ' + ah.Field);
    if (ah.field == 'MyDate__c') {
        Date oldValue = Date.valueOf(ah.OldValue);
        Date newValue = Date.valueOf(ah.NewValue);
    }
}

Versioned Behavior Changes
Date.valueOf has been versioned in these releases.
API version 33.0 or earlier
If you call `Date.valueOf` with a `Datetime` object, the method returns a `Date` value that contains the hours, minutes, seconds, and milliseconds set.

API version 34.0 to API version 53.0
If you call `Date.valueOf` with a `Datetime` object, the method converts `Datetime` to a valid `Date` without the time information, but the result depends on the manner in which the `Datetime` object was initialized. For example, if the `Datetime` object was initialized using `Datetime.valueOf(stringDate)`, the returned `Date` value contains time (hours) information. If the `Datetime` object is initialized using `Datetime.newInstance(year, month, day, hour, minute, second)` the returned `Date` value doesn't contain time information.

API version 54.0 and later
If you call `Date.valueOf` with a `Datetime` object, the method converts the object to a valid `Date` without the time information.

`year()`
Returns the year component of a `Date`

Signature
```
public Integer year()
```

Return Value
Type: `Integer`

Example
```
date myDate = date.newInstance(1988, 12, 17);
system.assertEquals(1988, myDate.year());
```

Datetime Class
Contains methods for the `Datetime` primitive data type.

Namespace
`System`

Usage
For more information about the `Datetime`, see `Datetime Data Type`.

Datetime Methods
The following are methods for `Datetime`. 
IN THIS SECTION:

addDays(additionalDays)
Adds the specified number of days to a Datetime.

addHours(additionalHours)
Adds the specified number of hours to a Datetime.

addMinutes(additionalMinutes)
Adds the specified number of minutes to a Datetime.

addMonths(additionalMonths)
Adds the specified number of months to a Datetime.

addSeconds(additionalSeconds)
Adds the specified number of seconds to a Datetime.

addYears(additionalYears)
Adds the specified number of years to a Datetime.

date()
Returns the Date component of a Datetime in the local time zone of the context user.

dateGMT()
Return the Date component of a Datetime in the GMT time zone.

day()
Returns the day-of-month component of a Datetime in the local time zone of the context user.

dayGmt()  
Returns the day-of-month component of a Datetime in the GMT time zone.

dayOfYear()
Returns the day-of-year component of a Datetime in the local time zone of the context user.

dayOfYearGmt()  
Returns the day-of-year component of a Datetime in the GMT time zone.

format()
Converts the date to the local time zone and returns the converted date as a formatted string using the locale of the context user. If the time zone cannot be determined, GMT is used.

format(dateFormatString)
Converts the date to the local time zone and returns the converted date as a string using the supplied Java simple date format. If the time zone cannot be determined, GMT is used.

format(dateFormatString, timezone)
Converts the date to the specified time zone and returns the converted date as a string using the supplied Java simple date format. If the supplied time zone is not in the correct format, GMT is used.

formatGmt(dateFormatString)
Returns a Datetime as a string using the supplied Java simple date format and the GMT time zone.

formatLong()
Converts the date to the local time zone and returns the converted date in long date format.

gTime()
Returns the number of milliseconds since January 1, 1970, 00:00:00 GMT represented by this DateTime object.
hour()
Returns the hour component of a Datetime in the local time zone of the context user.

hourGmt()
Returns the hour component of a Datetime in the GMT time zone.

isSameDay(dateToCompare)
Returns true if the Datetime that called the method is the same as the specified Datetime in the local time zone of the context user.

millisecond()
Returns the millisecond component of a Datetime in the local time zone of the context user.

millisecondGmt()
Returns the millisecond component of a Datetime in the GMT time zone.

minute()
Returns the minute component of a Datetime in the local time zone of the context user.

minuteGmt()
Returns the minute component of a Datetime in the GMT time zone.

month()
Returns the month component of a Datetime in the local time zone of the context user (1=Jan).

monthGmt()
Returns the month component of a Datetime in the GMT time zone (1=Jan).

newInstance(milliseconds)
Constructs a Datetime and initializes it to represent the specified number of milliseconds since January 1, 1970, 00:00:00 GMT.

newInstance(date, time)
Constructs a DateTime from the specified date and time in the local time zone.

newInstance(year, month, day)
Constructs a Datetime from Integer representations of the specified year, month (1=Jan), and day at midnight in the local time zone.

newInstance(year, month, day, hour, minute, second)
Constructs a Datetime from Integer representations of the specified year, month (1=Jan), hour, minute, and second in the local time zone.

newInstanceGmt(date, time)
Constructs a DateTime from the specified date and time in the GMT time zone.

newInstanceGmt(year, month, day)
Constructs a Datetime from Integer representations of the specified year, month (1=Jan), and day at midnight in the GMT time zone.

newInstanceGmt(year, month, day, hour, minute, second)
Constructs a Datetime from Integer representations of the specified year, month (1=Jan), day, hour, minute, and second in the GMT time zone.

now()
Returns the current Datetime based on a GMT calendar.

parse(datetimeString)
Constructs a Datetime from the given String in the local time zone and in the format of the user locale.

second()
Returns the second component of a Datetime in the local time zone of the context user.
secondGmt()
Returns the second component of a Datetime in the GMT time zone.

time()
Returns the time component of a Datetime in the local time zone of the context user.

timeGmt()
Returns the time component of a Datetime in the GMT time zone.

valueOf(dateTimeString)
Returns a Datetime that contains the value of the specified string.

valueOf(fieldValue)
Converts the specified object to a Datetime. Use this method to convert a history tracking field value or an object that represents a Datetime value.

valueOfGmt(dateTimeString)
Returns a Datetime that contains the value of the specified String.

year()
Returns the year component of a Datetime in the local time zone of the context user.

yearGmt()
Returns the year component of a Datetime in the GMT time zone.

addDays(additionalDays)
Adds the specified number of days to a Datetime.

Signature

public Datetime addDays(Integer additionalDays)

Parameters

additionalDays
Type: Integer

Return Value

Type: Datetime

Example

DateTime myDateTime = Datetime.newInstance(1960, 2, 17);
DateTime newDateTime = myDateTime.addDays(2);
DateTime expected = Datetime.newInstance(1960, 2, 19);
System.assertEquals(expected, newDateTime);

addHours(additionalHours)
Adds the specified number of hours to a Datetime.
Signature

public Datetime addHours(Integer additionalHours)

Parameters

additionalHours
Type: Integer

Return Value

Type: Datetime

Example

DateTime myDateTime = DateTime.newInstance(1997, 1, 31, 7, 8, 16);
DateTime newDateTime = myDateTime.addHours(3);
DateTime expected = DateTime.newInstance(1997, 1, 31, 10, 8, 16);
System.assertEquals(expected, newDateTime);

addMinutes(additionalMinutes)

Adds the specified number of minutes to a Datetime.

Signature

public Datetime addMinutes(Integer additionalMinutes)

Parameters

additionalMinutes
Type: Integer

Return Value

Type: Datetime

Example

DateTime myDateTime = DateTime.newInstance(1999, 2, 11, 8, 6, 16);
DateTime newDateTime = myDateTime.addMinutes(7);
DateTime expected = DateTime.newInstance(1999, 2, 11, 8, 13, 16);
System.assertEquals(expected, newDateTime);

addMonths(additionalMonths)

Adds the specified number of months to a Datetime.

Signature

public Datetime addMonths(Integer additionalMonths)
Parameters

`additionalMonths`

Type: `Integer`

Return Value

Type: `DateTime`

Example

```java
DateTime myDateTime = DateTime.newInstance(2000, 7, 7, 7, 8, 12);
DateTime newDateTime = myDateTime.addMonths(1);
DateTime expected = DateTime.newInstance(2000, 8, 7, 7, 8, 12);
System.assertEquals(expected, newDateTime);
```

`addSeconds(additionalSeconds)`

Adds the specified number of seconds to a Datetime.

Signature

```java
public Datetime addSeconds(Integer additionalSeconds)
```

Parameters

`additionalSeconds`

Type: `Integer`

Return Value

Type: `DateTime`

Example

```java
DateTime myDateTime = DateTime.newInstance(2001, 7, 19, 10, 7, 12);
DateTime newDateTime = myDateTime.addSeconds(4);
DateTime expected = DateTime.newInstance(2001, 7, 19, 10, 7, 16);
System.assertEquals(expected, newDateTime);
```

`addYears(additionalYears)`

Adds the specified number of years to a Datetime.

Signature

```java
public Datetime addYears(Integer additionalYears)
```
Parameters

\texttt{additionalYears}

Type: \texttt{Integer}

Return Value

Type: \texttt{Datetime}

Example

```java
DateTime myDateTime = DateTime.newInstance(2009, 12, 17, 13, 6, 6);
DateTime newDateTime = myDateTime.addYears(1);
DateTime expected = DateTime.newInstance(2010, 12, 17, 13, 6, 6);
System.assertEquals(expected, newDateTime);
```

\texttt{date()}

Returns the Date component of a Datetime in the local time zone of the context user.

Signature

\texttt{public Date date()}

Return Value

Type: \texttt{Date}

Example

```java
DateTime myDateTime = DateTime.newInstance(2006, 3, 16, 12, 6, 13);
Date myDate = myDateTime.date();
Date expected = Date.newInstance(2006, 3, 16);
System.assertEquals(expected, myDate);
```

\texttt{dateGMT()}

Return the Date component of a Datetime in the GMT time zone.

Signature

\texttt{public Date dateGMT()}

Return Value

Type: \texttt{Date}

Example

```java
// California local time, PST
DateTime myDateTime = DateTime.newInstance(2006, 3, 16, 23, 0, 0);
```
Date myDate = myDateTime.dateGMT();
Date expected = Date.newInstance(2006, 3, 17);
System.assertEquals(expected, myDate);

day()
Returns the day-of-month component of a Datetime in the local time zone of the context user.

Signature
public Integer day()

Return Value
Type: Integer

Example
DateTime myDateTime = DateTime.newInstance(1986, 2, 21, 23, 0, 0);
System.assertEquals(21, myDateTime.day());

dayGmt()
Returns the day-of-month component of a Datetime in the GMT time zone.

Signature
public Integer dayGmt()

Return Value
Type: Integer

Example
// California local time, PST
DateTime myDateTime = DateTime.newInstance(1987, 1, 14, 23, 0, 3);
System.assertEquals(15, myDateTime.dayGMT());

dayOfYear()
Returns the day-of-year component of a Datetime in the local time zone of the context user.

Signature
public Integer dayOfYear()

Return Value
Type: Integer
Example

For example, February 5, 2008 08:30:12 would be day 36.

```java
Datetime myDate = Datetime.newInstance(2008, 2, 5, 8, 30, 12);
system.assertEquals(myDate.dayOfYear(), 36);
```

dayOfYearGmt()

Returns the day-of-year component of a Datetime in the GMT time zone.

**Signature**

```java
public Integer dayOfYearGmt()
```

**Return Value**

Type: Integer

**Example**

```java
// This sample assumes we are in the PST timezone
DateTime myDateTime = DateTime.newInstance(1999, 2, 5, 23, 0, 3);
// January has 31 days + 5 days in February = 36 days
// dayOfYearGmt() adjusts the time zone from the current time zone to GMT
// by adding 8 hours to the PST time zone, so it's 37 days and not 36 days
System.assertEquals(37, myDateTime.dayOfYearGmt());
```

format()

Converts the date to the local time zone and returns the converted date as a formatted string using the locale of the context user. If the time zone cannot be determined, GMT is used.

**Signature**

```java
public String format()
```

**Return Value**

Type: String

**Example**

```java
Note: The sample is executed in an org where the “Enable ICU Locale Formats” crucial update is enabled. See https://releasenotes.docs.salesforce.com/en-us/spring20/release-notes/rn_forcecom_globalization_enable_icu_cruc.htm.

DateTime.myDateTime = DateTime.newInstance(1993, 6, 6, 3, 3, 3);
system.assertEquals('6/6/1993, 3:03 AM', mydatetime.format());
```
format(dateFormatString)
Converts the date to the local time zone and returns the converted date as a string using the supplied Java simple date format. If the time zone cannot be determined, GMT is used.

Signature
public String format(String dateFormatString)

Parameters
dateFormatString
Type: String

Return Value
Type: String

Usage
For more information on the Java simple date format, see Java SimpleDateFormat.

Example
```
Datetime myDT = DateTime.newInstance(2022, 5, 4, 19, 37, 55);
String myDate = myDT.format('yyyy-MM-dd h:mm a');
String expected = '2022-05-04 7:37 PM';
System.assertEquals(expected, myDate);
```

format(dateFormatString, timezone)
Converts the date to the specified time zone and returns the converted date as a string using the supplied Java simple date format. If the supplied time zone is not in the correct format, GMT is used.

Signature
public String format(String dateFormatString, String timezone)

Parameters
dateFormatString
Type: String
timezone
Type: String

Valid time zone values for the timezone argument are the time zones of the Java TimeZone class that correspond to the time zones returned by the TimeZone.getAvailableIDs method in Java. We recommend you use full time zone names, not the three-letter abbreviations.
**Return Value**
Type: `String`

**Usage**
For more information on the Java simple date format, see Java `SimpleDateFormat`.

**Example**
This example uses `format` to convert a GMT date to the America/New_York time zone and formats the date using the specified date format.

```java
Datetime GMTDate = Datetime.newInstanceGmt(2011, 6, 1, 12, 1, 5);
String strConvertedDate = GMTDate.format('MM/dd/yyyy HH:mm:ss', 'America/New_York');
// Date is converted to
// the new time zone and is adjusted
// for daylight saving time.
System.assertEquals('06/01/2011 08:01:05', strConvertedDate);
```

**formatGmt(dateFormatString)**
Returns a Datetime as a string using the supplied Java simple date format and the GMT time zone.

**Signature**
```java
public String formatGmt(String dateFormatString)
```

**Parameters**
`dateFormatString`
Type: `String`

**Return Value**
Type: `String`

**Usage**
For more information on the Java simple date format, see Java `SimpleDateFormat`.

**Example**
```java
DateTime myDateTime = DateTime.newInstance(1993, 6, 6, 3, 3, 3);
String formatted = myDateTime.formatGMT('EEE, MMM d yyyy HH:mm:ss');
String expected = 'Sun, Jun 6 1993 10:03:03';
System.assertEquals(expected, formatted);
```
**formatLong()**
Converting the date to the local time zone and returns the converted date in long date format.

**Signature**
```
public String formatLong()
```

**Return Value**
Type: String

**Example**
```java
// Passing local date based on the PST time zone
Datetime dt = DateTime.newInstance(2012,12,28,10,0,0);
// Writes 12/28/2012 10:00:00 AM PST
System.debug('dt.formatLong()=' + dt.formatLong());
```

**getTime()**
Returns the number of milliseconds since January 1, 1970, 00:00:00 GMT represented by this DateTime object.

**Signature**
```
public Long getTime()
```

**Return Value**
Type: Long

**Example**
```java
DateTime dt = DateTime.newInstance(2007, 6, 23, 3, 3, 3);
Long gettime = dt.getTime();
Long expected = 1182592983000L;
System.assertEquals(expected, gettime);
```

**hour()**
Returns the hour component of a Datetime in the local time zone of the context user.

**Signature**
```
public Integer hour()
```

**Return Value**
Type: Integer
Example

```java
DateTime myDateTime = DateTime.newInstance(1998, 11, 21, 3, 3, 3);
System.assertEquals(3 , myDateTime.hour());
```

`hourGmt()`

Returns the hour component of a Datetime in the GMT time zone.

**Signature**

```java
public Integer hourGmt()
```

**Return Value**

Type: `Integer`

Example

```java
// California local time
DateTime myDateTime = DateTime.newInstance(2000, 4, 27, 3, 3, 3);
System.assertEquals(10 , myDateTime.hourGmt());
```

`isSameDay(dateToCompare)`

Returns true if the Datetime that called the method is the same as the specified Datetime in the local time zone of the context user.

**Signature**

```java
public Boolean isSameDay(Datetime dateToCompare)
```

**Parameters**

`dateToCompare`  
Type: `Datetime`

**Return Value**

Type: `Boolean`

Example

```java
datetime myDate = datetime.now();
datetime dueDate =
  datetime.newInstance(2008, 1, 30);
boolean dueNow = myDate.isSameDay(dueDate);
```

`millisecond()`

Return the millisecond component of a Datetime in the local time zone of the context user.
Signature

public Integer millisecond()

Return Value
Type: Integer

Example

DateTime myDateTime = DateTime.now();
system.debug(myDateTime.millisecond());

millisecondGmt()
Return the millisecond component of a DateTime in the GMT time zone.

Signature

public Integer millisecondGmt()

Return Value
Type: Integer

Example

DateTime myDateTime = DateTime.now();
system.debug(myDateTime.millisecondGMT());

minute()
Returns the minute component of a DateTime in the local time zone of the context user.

Signature

public Integer minute()

Return Value
Type: Integer

Example

DateTime myDateTime = DateTime.newInstance(2001, 2, 27, 3, 3, 3);
system.assertEquals(3, myDateTime.minute());

minuteGmt()
Returns the minute component of a DateTime in the GMT time zone.

Signature

public Integer minuteGmt()
Signature

public Integer minuteGmt()

Return Value
Type: Integer

Example

```java
DateTime myDateTime = DateTime.newInstance(2002, 12, 3, 3, 3, 3);
System.assertEquals(3, myDateTime.minuteGMT());
```

**month()**

Returns the month component of a Datetime in the local time zone of the context user (1=Jan).

Signature

public Integer month()

Return Value
Type: Integer

Example

```java
DateTime myDateTime = DateTime.newInstance(2004, 11, 4, 3, 3, 3);
System.assertEquals(11, myDateTime.month());
```

**monthGmt()**

Returns the month component of a Datetime in the GMT time zone (1=Jan).

Signature

public Integer monthGmt()

Return Value
Type: Integer

Example

```java
DateTime myDateTime = DateTime.newInstance(2006, 11, 19, 3, 3, 3);
System.assertEquals(11, myDateTime.monthGmt());
```

**newInstance(milliseconds)**

Constructs a Datetime and initializes it to represent the specified number of milliseconds since January 1, 1970, 00:00:00 GMT.
Signature

public static Datetime newInstance(Long milliseconds)

Parameters

milliseconds
Type: Long

Return Value

Type: Datetime
The returned date is in the GMT time zone.

Example

```
Long longtime = 1341828183000L;
DateTime dt = DateTime.newInstance(longtime);
DateTime expected = DateTime.newInstance(2012, 7, 09, 3, 3, 3);
System.assertEquals(expected, dt);
```

newInstance(date, time)
Constructs a DateTime from the specified date and time in the local time zone.

Signature

public static Datetime newInstance(Date date, Time time)

Parameters

date
Type: Date
time
Type: Time

Return Value

Type: Datetime
The returned date is in the GMT time zone.

Example

```
Date myDate = Date.newInstance(2011, 11, 18);
Time myTime = Time.newInstance(3, 3, 3, 0);
DateTime dt = DateTime.newInstance(myDate, myTime);
DateTime expected = DateTime.newInstance(2011, 11, 18, 3, 3, 3);
System.assertEquals(expected, dt);
```
**newInstance(year, month, day)**

Constructs a Datetime from Integer representations of the specified year, month (1=Jan), and day at midnight in the local time zone.

**Signature**

```java
public static Datetime newInstance(Integer year, Integer month, Integer day)
```

**Parameters**

- **year**
  - Type: Integer
- **month**
  - Type: Integer
- **day**
  - Type: Integer

**Return Value**

Type: Datetime

The returned date is in the GMT time zone.

**Example**

```java
datetime myDate = datetime.newInstance(2008, 12, 1);
```

**newInstance(year, month, day, hour, minute, second)**

Constructs a Datetime from Integer representations of the specified year, month (1=Jan), day, hour, minute, and second in the local time zone.

**Signature**

```java
public static Datetime newInstance(Integer year, Integer month, Integer day, Integer hour, Integer minute, Integer second)
```

**Parameters**

- **year**
  - Type: Integer
- **month**
  - Type: Integer
- **day**
  - Type: Integer
- **hour**
  - Type: Integer
- **minute**
  - Type: Integer
- **second**
  - Type: Integer
second
Type: Integer

Return Value
Type: Datetime
The returned date is in the GMT time zone.

Example
```
Datetime myDate = Datetime.newInstance(2008, 12, 1, 12, 30, 2);
```

newInstanceGmt(date, time)
Constructs a DateTime from the specified date and time in the GMT time zone.

Signature
```
public static Datetime newInstanceGmt(Date date, Time time)
```

Parameters
- date  
  Type: Date
- time  
  Type: Time

Return Value
Type: Datetime

Example
```
Date myDate = Date.newInstance(2013, 11, 12);
Time myTime = Time.newInstance(3, 3, 3, 0);
DateTime dt = DateTime.newInstanceGMT(myDate, myTime);
DateTime expected = DateTime.newInstanceGMT(2013, 11, 12, 3, 3, 3);
System.assertEquals(expected, dt);
```

newInstanceGmt(year, month, date)
Constructs a Datetime from Integer representations of the specified year, month (1=Jan), and day at midnight in the GMT time zone.

Signature
```
public static Datetime newInstanceGmt(Integer year, Integer month, Integer date)
```
Parameters

*year*
  Type: Integer

*month*
  Type: Integer

$date*$
  Type: Integer

Return Value

Type: Datetime

Example

```java
DateTime dt = DateTime.newInstanceGMT(1996, 3, 22);
```

**newInstanceGmt**(*year*, *month*, *date*, *hour*, *minute*, *second*)

Constructs a Datetime from Integer representations of the specified year, month (1=Jan), day, hour, minute, and second in the GMT time zone

Signature

```java
public static Datetime newInstanceGmt(Integer year, Integer month, Integer date, Integer hour, Integer minute, Integer second)
```

Parameters

*year*
  Type: Integer

*month*
  Type: Integer

$date*$
  Type: Integer

*hour*
  Type: Integer

*minute*
  Type: Integer

*second*
  Type: Integer

Return Value

Type: Datetime
Example

```
//California local time
DateTime dt = DateTime.newInstanceGMT(1998, 1, 29, 2, 2, 3);
DateTime expected = DateTime.newInstance(1998, 1, 28, 18, 2, 3);
System.assertEquals(expected, dt);
```

`now()`

Returns the current Datetime based on a GMT calendar.

Signature

```
public static Datetime now()
```

Return Value

Type: Datetime

The format of the returned datetime is: 'MM/DD/YYYY HH:MM PERIOD'

Example

```
datetime myDateTime = datatime.now();
```

`parse(datatimeString)`

Constructs a Datetime from the given String in the local time zone and in the format of the user locale.

Signature

```
public static Datetime parse(String datatimeString)
```

Parameters

```
datatimeString
    Type: String
```

Return Value

Type: Datetime

The returned date is in the GMT time zone.

Example

This example uses parse to create a Datetime from a date passed in as a string and that is formatted for the English (United States) locale. You may need to change the format of the date string if you have a different locale.
Note: This sample is executed in an org where the “Enable ICU Locale Formats” crucial update is enabled. See https://releasenotes.docs.salesforce.com/en-us/spring20/release-notes/rn_forcecom_globalization_enable_icu_cruc.htm.

```java
Datetime dt = DateTime.parse('10/14/2011, 11:46 AM');
String myDtString = dt.format();
system.assertEquals(myDtString, '10/14/2011, 11:46 AM');
```

**second()**

Returns the second component of a Datetime in the local time zone of the context user.

**Signature**

```java
public Integer second()
```

**Return Value**

Type: Integer

**Example**

```java
DateTime dt = DateTime.newInstanceGMT(1999, 9, 22, 3, 1, 2);
System.assertEquals(2, dt.second());
```

**secondGmt()**

Returns the second component of a Datetime in the GMT time zone.

**Signature**

```java
public Integer secondGmt()
```

**Return Value**

Type: Integer

**Example**

```java
DateTime dt = DateTime.newInstance(2000, 2, 3, 3, 1, 5);
System.assertEquals(5, dt.secondGMT());
```

**time()**

Returns the time component of a Datetime in the local time zone of the context user.

**Signature**

```java
public Time time()
```
Return Value
Type: Time

Example
```java
DateTime dt = DateTime.newInstance(2002, 11, 21, 0, 2, 2);
Time expected = Time.newInstance(0, 2, 2, 0);
System.assertEquals(expected, dt.time());
```

`timeGmt()`
Returns the time component of a Datetime in the GMT time zone.

Signature
```java
public Time timeGmt()
```

Return Value
Type: Time

Example
```java
// This sample is based on the PST time zone
DateTime dt = DateTime.newInstance(2004, 1, 27, 4, 1, 2);
Time expected = Time.newInstance(12, 1, 2, 0);
// 8 hours are added to the time to convert it from
// PST to GMT
System.assertEquals(expected, dt.timeGMT());
```

`valueOf(dateTimeString)`
Returns a Datetime that contains the value of the specified string.

Signature
```java
public static Datetime valueOf(String dateTimeString)
```

Parameters
- `dateTimeString`  
  Type: `String`

Return Value
Type: `Datetime`  
The returned date is in the GMT time zone.
Usage

The specified string should use the standard date format "yyyy-MM-dd HH:mm:ss" in the local time zone.

Example

```java
string year = '2008';
string month = '10';
string day = '5';
string hour = '12';
string minute = '20';
string second = '20';
string stringDate = year + '-' + month + '-' + day + ' ' + hour + ':'
    + minute + ':' + second;
Datetime myDate = Datetime.valueOf(stringDate);
```

valueOf(fieldValue)

Converts the specified object to a Datetime. Use this method to convert a history tracking field value or an object that represents a Datetime value.

Signature

```java
public static Datetime valueOf(Object fieldValue)
```

Parameters

- **fieldValue**
  
  Type: Object

Return Value

Type: Datetime

Usage

Use this method with the **OldValue** or **NewValue** fields of history sObjects, such as **AccountHistory**, when the field is a Date/Time field.

Example

```java
List<AccountHistory> ahlist = [SELECT Field,OldValue,NewValue FROM AccountHistory];
for(AccountHistory ah : ahlist) {
    System.debug('Field: ' + ah.Field);
    if (ah.field == 'MyDatetime__c') {
        Datetime oldValue = Datetime.valueOf(ah.OldValue);
        Datetime newValue = Datetime.valueOf(ah.NewValue);
    }
}
```
valueOfGmt(dateTimeString)
Returns a Datetime that contains the value of the specified String.

Signature

public static Datetime valueOfGmt(String dateTimeString)

Parameters

datTimeString
Type: String

Return Value

Type: Datetime

Usage

The specified string should use the standard date format "yyyy-MM-dd HH:mm:ss" in the GMT time zone.

Example

// California locale time
string year = '2009';
string month = '3';
string day = '5';
string hour = '5';
string minute = '2';
string second = '2';
string stringDate = year + '-' + month + '-' + day + ' ' + hour + ':'
   + minute + ':' + second;

DateTime myDate = Datetime.valueOfGMT(stringDate);

DateTime expected = DateTime.newInstance(2009, 3, 4, 21, 2, 2);
System.assertEquals(expected, myDate);

year()

Returns the year component of a Datetime in the local time zone of the context user.

Signature

public Integer year()
Example

```java
DateTime dt = DateTime.newInstance(2012, 1, 26, 5, 2, 4);
System.assertEquals(2012, dt.year());
```

```
yearGmt()
```

Returns the year component of a Datetime in the GMT time zone.

Signature

```
public Integer yearGmt()
```

Return Value

Type: Integer

Example

```java
DateTime dt = DateTime.newInstance(2012, 10, 4, 6, 4, 6);
System.assertEquals(2012, dt.yearGMT());
```

Decimal Class

Contains methods for the Decimal primitive data type.

Namespace

```
System
```

Usage

Note: Two Decimal objects that are numerically equivalent but differ in scale (such as 1.1 and 1.10) generally do not have the same hashcode. Use caution when such Decimal objects are used in Sets or as Map keys.

For more information on Decimal, see Decimal Data Type.

IN THIS SECTION:

Rounding Mode

Rounding mode specifies the rounding behavior for numerical operations capable of discarding precision.

Decimal Methods

Rounding Mode

Rounding mode specifies the rounding behavior for numerical operations capable of discarding precision.

Each rounding mode indicates how the least significant returned digit of a rounded result is to be calculated. The following are the valid values for `roundingMode`. 
<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| CEILING| Rounds towards positive infinity. That is, if the result is positive, this mode behaves the same as the **UP** rounding mode; if the result is negative, it behaves the same as the **DOWN** rounding mode. Note that this rounding mode never decreases the calculated value. For example:  
  - Input number 5.5: **CEILING** round mode result: 6  
  - Input number 1.1: **CEILING** round mode result: 2  
  - Input number -1.1: **CEILING** round mode result: -1  
  - Input number -2.7: **CEILING** round mode result: -2  
  ```java
  Decimal[] example = new Decimal[]{5.5, 1.1, -1.1, -2.7};
  Long[] expected = new Long[]{6, 2, -1, -2};
  for(integer x = 0; x < example.size(); x++){
      System.assertEquals(expected[x],
                           example[x].round(System.RoundingMode.CEILING));
  }
  ```                                                                                                                                                                                                                                                                                                                                                                      |
| DOWN   | Rounds towards zero. This rounding mode always discards any fractions (decimal points) prior to executing. Note that this rounding mode never increases the magnitude of the calculated value. For example:  
  - Input number 5.5: **DOWN** round mode result: 5  
  - Input number 1.1: **DOWN** round mode result: 1  
  - Input number -1.1: **DOWN** round mode result: -1  
  - Input number -2.7: **DOWN** round mode result: -2  
  ```java
  Decimal[] example = new Decimal[]{5.5, 1.1, -1.1, -2.7};
  Long[] expected = new Long[]{5, 1, -1, -2};
  for(integer x = 0; x < example.size(); x++){
      System.assertEquals(expected[x],
                           example[x].round(System.RoundingMode.DOWN));
  }
  ```                                                                                                                                                                                                                                                                                                                                                                      |
| FLOOR  | Rounds towards negative infinity. That is, if the result is positive, this mode behaves the same as the **DOWN** rounding mode; if negative, this mode behaves the same as the **UP** rounding mode. Note that this rounding mode never increases the calculated value. For example:  
  - Input number 5.5: **FLOOR** round mode result: 5  
  - Input number 1.1: **FLOOR** round mode result: 1  
  - Input number -1.1: **FLOOR** round mode result: -2  
  - Input number -2.7: **FLOOR** round mode result: -3  
  ```java
  Decimal[] example = new Decimal[]{5.5, 1.1, -1.1, -2.7};
  Long[] expected = new Long[]{5, 1, -2, -3};
  for(integer x = 0; x < example.size(); x++){
      System.assertEquals(expected[x],
                           example[x].round(System.RoundingMode.FLOOR));
  }
  ```                                                                                                                                                                                                                                                                                                                                                                      |
### HALF_DOWN

Rounds towards the “nearest neighbor” unless both neighbors are equidistant, in which case this mode rounds down. This rounding mode behaves the same as the UP rounding mode if the discarded fraction (decimal point) is > 0.5; otherwise, it behaves the same as DOWN rounding mode. For example:

- Input number 5.5: HALF_DOWN round mode result: 5
- Input number 1.1: HALF_DOWN round mode result: 1
- Input number -1.1: HALF_DOWN round mode result: -1
- Input number -2.7: HALF_DOWN round mode result: -3

```java
decimal example = new decimal[]{5.5, 1.1, -1.1, -2.7};
long[] expected = new long[]{5, 1, -1, -3};
for (integer x = 0; x < example.size(); x++) {
    system.assertEquals(expected[x],
        example[x].round(system.roundingmode.half_down));
}
```

### HALF_EVEN

Rounds towards the “nearest neighbor” unless both neighbors are equidistant, in which case, this mode rounds towards the even neighbor. This rounding mode behaves the same as the HALF_UP rounding mode if the digit to the left of the discarded fraction (decimal point) is odd. It behaves the same as the HALF_DOWN rounding method if it is even. For example:

- Input number 5.5: HALF_EVEN round mode result: 6
- Input number 1.1: HALF_EVEN round mode result: 1
- Input number -1.1: HALF_EVEN round mode result: -1
- Input number -2.7: HALF_EVEN round mode result: -3

```java

decimal[] example = new decimal[](5.5, 1.1, -1.1, -2.7);
long[] expected = new long[]{6, 1, -1, -3};
for (integer x = 0; x < example.size(); x++) {
    system.assertEquals(expected[x],
        example[x].round(system.roundingmode.half_even));
}
```

Note that this rounding mode statistically minimizes cumulative error when applied repeatedly over a sequence of calculations.

### HALF_UP

Rounds towards the “nearest neighbor” unless both neighbors are equidistant, in which case, this mode rounds up. This rounding method behaves the same as the UP rounding method if the discarded fraction (decimal point) is >= 0.5; otherwise, this rounding method behaves the same as the DOWN rounding method. For example:

- Input number 5.5: HALF_UP round mode result: 6
- Input number 1.1: HALF_UP round mode result: 1
- Input number -1.1: HALF_UP round mode result: -1
- Input number -2.7: HALF_UP round mode result: -3

```java

decimal[] example = new decimal[](5.5, 1.1, -1.1, -2.7);
long[] expected = new long[]{6, 1, -1, -3};
for (integer x = 0; x < example.size(); x++) {
    system.assertEquals(expected[x],
        example[x].round(system.roundingmode.half_up));
}
```
**UNNECESSARY**

Asserts that the requested operation has an exact result, which means that no rounding is necessary. If this rounding mode is specified on an operation that yields an inexact result, a `MathException` is thrown. For example:

- Input number 5.5: **UNNECESSARY** round mode result: `MathException`
- Input number 1.1: **UNNECESSARY** round mode result: `MathException`
- Input number 1.0: **UNNECESSARY** round mode result: 1
- Input number -1.0: **UNNECESSARY** round mode result: -1
- Input number -2.2: **UNNECESSARY** round mode result: `MathException`

```java
Decimal example1 = 5.5;
Decimal example2 = 1.0;
system.assertEquals(1,
    example2.round(System.RoundingMode.UNNECESSARY));
try{
    example1.round(System.RoundingMode.UNNECESSARY);
} catch(Exception E) {
    system.assertEquals('System.MathException', E.getTypeName());
}
```

**UP**

Rounds away from zero. This rounding mode always truncates any fractions (decimal points) prior to executing. Note that this rounding mode never decreases the magnitude of the calculated value. For example:

- Input number 5.5: **UP** round mode result: 6
- Input number 1.1: **UP** round mode result: 2
- Input number -1.1: **UP** round mode result: -2
- Input number -2.7: **UP** round mode result: -3

```java
Decimal[] example = new Decimal[]{5.5, 1.1, -1.1, -2.7};
Long[] expected = new Long[]{6, 2, -2, -3};
for(integer x = 0; x < example.size(); x++){
    System.assertEquals(expected[x],
        example[x].round(System.RoundingMode.UP));
}
```

---

**Decimal Methods**

The following are methods for `Decimal`.

**IN THIS SECTION:**

`abs()`

Returns the absolute value of the `Decimal`.

---

3042
divide(divisor, scale)
Divides this Decimal by the specified divisor, and sets the scale, that is, the number of decimal places, of the result using the specified scale.

divide(divisor, scale, roundingMode)
Divides this Decimal by the specified divisor, sets the scale, that is, the number of decimal places, of the result using the specified scale, and if necessary, rounds the value using the rounding mode.

doubleValue()
Returns the Double value of this Decimal.

format()
Returns the String value of this Decimal using the locale of the context user.

intValue()
Returns the Integer value of this Decimal.

longValue()
Returns the Long value of this Decimal.

pow(exponent)
Returns the value of this decimal raised to the power of the specified exponent.

precision()
Returns the total number of digits for the Decimal.

round()
Returns the rounded approximation of this Decimal. The number is rounded to zero decimal places using half-even rounding mode, that is, it rounds towards the “nearest neighbor” unless both neighbors are equidistant, in which case, this mode rounds towards the even neighbor.

round(roundingMode)
Returns the rounded approximation of this Decimal. The number is rounded to zero decimal places using the rounding mode specified by the rounding mode.

scale()
Returns the scale of the Decimal, that is, the number of decimal places.

setScale(scale)
Returns the Decimal scaled to the specified number of decimal places, using half-even rounding, if necessary. Half-even rounding mode rounds toward the “nearest neighbor.” If both neighbors are equidistant, the number is rounded toward the even neighbor.

setScale(scale, roundingMode)
Returns the Decimal scaled to the specified number of decimal places, using the specified rounding mode, if necessary.

stripTrailingZeros()
Returns the Decimal with any trailing zeros removed.

toPlainString()
Returns the String value of this Decimal, without using scientific notation.

valueOf(doubleToDecimal)
Returns a Decimal that contains the value of the specified Double.

valueOf(longToDecimal)
Returns a Decimal that contains the value of the specified Long.
valueOf(stringToDecimal)
Returns a Decimal that contains the value of the specified String. As in Java, the string is interpreted as representing a signed Decimal.

abs()
Returns the absolute value of the Decimal.

Signature
public Decimal abs()

Return Value
Type: Decimal

Example
Decimal myDecimal = -6.02214129;
System.assertEquals(6.02214129, myDecimal.abs());

divide(divisor, scale)
Divides this Decimal by the specified divisor, and sets the scale, that is, the number of decimal places, of the result using the specified scale.

Signature
public Decimal divide(Decimal divisor, Integer scale)

Parameters
divisor
Type: Decimal
scale
Type: Integer

Return Value
Type: Decimal

Example
Decimal decimalNumber = 19;
Decimal result = decimalNumber.divide(100, 3);
System.assertEquals(0.190, result);

divide(divisor, scale, roundingMode)
Divides this Decimal by the specified divisor, sets the scale, that is, the number of decimal places, of the result using the specified scale, and if necessary, rounds the value using the rounding mode.
Decide Class

**Signature**

public Decimal divide(Decimal divisor, Integer scale, System.RoundingMode roundingMode)

**Parameters**

divisor
  Type: Decimal

scale
  Type: Integer

roundingMode
  Type: System.RoundingMode

**Return Value**

Type: Decimal

**Example**

```java
Decimal myDecimal = 12.4567;
Decimal divDec = myDecimal.divide(7, 2, System.RoundingMode.UP);
System.assertEquals(divDec, 1.78);
```

**doubleValue()**

Returns the Double value of this Decimal.

**Signature**

public Double doubleValue()

**Return Value**

Type: Double

**Example**

```java
Decimal myDecimal = 6.62606957;
Double value = myDecimal.doubleValue();
System.assertEquals(6.62606957, value);
```

**format()**

Returns the String value of this Decimal using the locale of the context user.

**Signature**

public String format()
Return Value
Type: String

Usage
Scientific notation will be used if an exponent is needed.

Example
```java
// U.S. locale
Decimal myDecimal = 12345.6789;
system.assertEquals('12,345.679', myDecimal.format());
```

`intValue()`
Returns the Integer value of this Decimal.

Signature
```java
public Integer intValue()
```

Return Value
Type: Integer

Example
```java
Decimal myDecimal = 1.602176565;
system.assertEquals(1, myDecimal.intValue());
```

`longValue()`
Returns the Long value of this Decimal.

Signature
```java
public Long longValue()
```

Return Value
Type: Long

Example
```java
Decimal myDecimal = 376.730313461;
system.assertEquals(376, myDecimal.longValue());
```

`pow(exponent)`
Returns the value of this decimal raised to the power of the specified exponent.
Signature

public Decimal pow(Integer exponent)

Parameters

exponent
Type: Integer
The value of exponent must be between 0 and 32,767.

Return Value
Type: Decimal

Usage
If you use MyDecimal.pow(0), 1 is returned.
The Math.pow method does accept negative values.

Example

Decimal myDecimal = 4.12;
Decimal powDec = myDecimal.pow(2);
System.assertEquals(powDec, 16.9744);

precision()

Returns the total number of digits for the Decimal.

Signature

public Integer precision()

Return Value
Type: Integer

Example

For example, if the Decimal value was 123.45, precision returns 5. If the Decimal value is 123.123, precision returns 6.

Decimal D1 = 123.45;
Integer precision1 = D1.precision();
System.assertEquals(precision1, 5);
Decimal D2 = 123.123;
Integer precision2 = D2.precision();
System.assertEquals(precision2, 6);
round()

Returns the rounded approximation of this Decimal. The number is rounded to zero decimal places using half-even rounding mode, that is, it rounds towards the “nearest neighbor” unless both neighbors are equidistant, in which case, this mode rounds towards the even neighbor.

**Signature**

```java
public Long round()
```

**Return Value**

Type: `Long`

**Usage**

Note that this rounding mode statistically minimizes cumulative error when applied repeatedly over a sequence of calculations.

**Example**

```java
Decimal D = 4.5;
Long L = D.round();
System.assertEquals(4, L);

Decimal D1 = 5.5;
Long L1 = D1.round();
System.assertEquals(6, L1);

Decimal D2 = 5.2;
Long L2 = D2.round();
System.assertEquals(5, L2);

Decimal D3 = -5.7;
Long L3 = D3.round();
System.assertEquals(-6, L3);
```

round(roundingMode)

Returns the rounded approximation of this Decimal. The number is rounded to zero decimal places using the rounding mode specified by the rounding mode.

**Signature**

```java
public Long round(System.RoundingMode roundingMode)
```

**Parameters**

- **roundingMode**
  Type: `System.RoundingMode`
Return Value
Type: Long

scale()
Returns the scale of the Decimal, that is, the number of decimal places.

Signature
public Integer scale()

Return Value
Type: Integer

Example
Decimal myDecimal = 9.27400968;
system.assertEquals(8, myDecimal.scale());

setScale(scale)
Returns the Decimal scaled to the specified number of decimal places, using half-even rounding, if necessary. Half-even rounding mode rounds toward the “nearest neighbor.” If both neighbors are equidistant, the number is rounded toward the even neighbor.

Signature
public Decimal setScale(Integer scale)

Parameters
scale
Type: Integer
The value of scale must be between –33 and 33. If the value of scale is negative, your unscaled value is multiplied by 10 to the power of the negation of scale. For example, after this operation, the value of d is $4 \times 10^{-3}$.

Decimal d = 4000;
d = d.setScale(-3);

Return Value
Type: Decimal

Usage
If you do not explicitly set the scale for a Decimal, the item from which the Decimal is created determines the scale.

- If the Decimal is created as part of a query, the scale is based on the scale of the field returned from the query.
- If the Decimal is created from a String, the scale is the number of characters after the decimal point of the String.
If the Decimal is created from a non-decimal number, the number is first converted to a String. Scale is then set using the number of characters after the decimal point.

Example

```java
Decimal myDecimal = 8.987551787;
Decimal setScaled = myDecimal.setScale(3);
System.assertEquals(8.988, setScaled);
```

**setScale(scale, roundingMode)**

Returns the Decimal scaled to the specified number of decimal places, using the specified rounding mode, if necessary.

**Signature**

```java
public Decimal setScale(Integer scale, System.RoundingMode roundingMode)
```

**Parameters**

- `scale`
  - Type: `Integer`
  - The value of `scale` must be between -33 and 33. If the value of `scale` is negative, your unscaled value is multiplied by 10 to the power of the negation of `scale`. For example, after this operation, the value of `d` is $4 \times 10^{-3}$.

  ```java
  Decimal d = 4000;
  d = d.setScale(-3);
  ```

- `roundingMode`
  - Type: `System.RoundingMode`

**Return Value**

Type: `Decimal`

**Usage**

If you do not explicitly set the scale for a Decimal, the item from which the Decimal is created determines the scale.

- If the Decimal is created as part of a query, the scale is based on the scale of the field returned from the query.
- If the Decimal is created from a String, the scale is the number of characters after the decimal point of the String.
- If the Decimal is created from a non-decimal number, the number is first converted to a String. Scale is then set using the number of characters after the decimal point.

**stripTrailingZeros()**

Returns the Decimal with any trailing zeros removed.

**Signature**

```java
public Decimal stripTrailingZeros()
```
Return Value
Type: Decimal

Example
```java
Decimal myDecimal = 1.10000;
Decimal stripped = myDecimal.stripTrailingZeros();
System.assertEquals(stripped, 1.1);
```

toPlainString()
Returns the String value of this Decimal, without using scientific notation.

Signature
```java
public String toPlainString()
```

Return Value
Type: String

Example
```java
Decimal myDecimal = 12345.6789;
System.assertEquals('12345.6789', myDecimal.toPlainString());
```

valueOf(doubleToDecimal)
Returns a Decimal that contains the value of the specified Double.

Signature
```java
public static Decimal valueOf(Double doubleToDecimal)
```

Parameters
doubleToDecimal
  Type: Double

Return Value
Type: Decimal

Example
```java
Double myDouble = 2.718281828459045;
Decimal myDecimal = Decimal.valueOf(myDouble);
System.assertEquals(2.718281828459045, myDecimal);
```
**valueOf (longToDecimal)**

Returns a Decimal that contains the value of the specified Long.

**Signature**

```java
public static Decimal valueOf(Long longToDecimal)
```

**Parameters**

- `longToDecimal`
  
  Type: Long

**Return Value**

Type: Decimal

**Example**

```java
Long myLong = 299792458;
Decimal myDecimal = Decimal.valueOf(myLong);
System.assertEquals(299792458, myDecimal);
```

**valueOf (stringToDecimal)**

Returns a Decimal that contains the value of the specified String. As in Java, the string is interpreted as representing a signed Decimal.

**Signature**

```java
public static Decimal valueOf(String stringToDecimal)
```

**Parameters**

- `stringToDecimal`
  
  Type: String

**Return Value**

Type: Decimal

**Example**

```java
String temp = '12.4567';
Decimal myDecimal = Decimal.valueOf(temp);
```

**Domain Class**

Represents an existing domain hosted by Salesforce that serves the org or its content. Contains methods to obtain information about these domains, such as the domain type, My Domain name, and sandbox name.
Namespace

System

Usage

Use the Domain class to obtain information about the domains that Salesforce hosts for your org. This class only applies to domains hosted by Salesforce, and can’t be used to generate a new domain.

Example

This code uses the System.DomainParser class to parse a hostname. It then gets the associated domain type.

```java
System.Domain d = DomainParser.parse('mycompany.lightning.force.com');
String myDomainName = d.getMyDomainName();
System.DomainType domainType = d.getDomainType();
```

IN THIS SECTION:

Domain Methods

Domain Methods

The following are methods for Domain.

IN THIS SECTION:

getDomainType()

Returns the domain’s type, such as CONTENT_DOMAIN, EXPERIENCE_CLOUD_SITES_DOMAIN, or LIGHTNING_DOMAIN.

getMyDomainName()

Returns the domain’s My Domain name.

getPackageName()

For a domain that includes the package name, such as a Lightning Component domain or Visualforce page domain, returns the package name. For a domain that doesn’t contain a package name, this method returns null.

g getSandboxName()

For a sandbox org domain, returns the sandbox name. For a production org domain, returns null.

g getSitesSubdomainName()

For a system-managed Experience Cloud site domain or Salesforce Site domain, returns the sites subdomain name. If enhanced domains are enabled, this method always returns null. When enhanced domains are enabled, the org’s My Domain name is the subdomain for the system-managed domains for Experience Cloud sites and Salesforce Sites domains.

getDomainType()

Returns the domain’s type, such as CONTENT_DOMAIN, EXPERIENCE_CLOUD_SITES_DOMAIN, or LIGHTNING_DOMAIN.

Signature

```java
public System.DomainType getDomainType()
```
Return Value
Type: System.DomainType

getMyDomainName()
Returns the domain's My Domain name.

Signature
public String getMyDomainName()

Return Value
Type: String

getPackageName()
For a domain that includes the package name, such as a Lightning Component domain or Visualforce page domain, returns the package name. For a domain that doesn't contain a package name, this method returns null.

Signature
public String getPackageName()

Return Value
Type: String

getSandboxName()
For a sandbox org domain, returns the sandbox name. For a production org domain, returns null.

Signature
public String getSandboxName()

Return Value
Type: String

getSitesSubdomainName()
For a system-managed Experience Cloud site domain or Salesforce Site domain, returns the sites subdomain name. If enhanced domains are enabled, this method always returns null. When enhanced domains are enabled, the org's My Domain name is the subdomain for the system-managed domains for Experience Cloud sites and Salesforce Sites domains.

Signature
public String getSitesSubdomainName()
Return Value
Type: String

DomainCreator Class
Use the DomainCreator class to return a hostname specific to the org. For example, get the org’s Visualforce hostname. Values are returned as a hostname, such as `MyDomainName.lightning.force.com`.

Namespace
System

Examples

This example code fetches the org’s My Domain login hostname and the Visualforce hostname for the `uat` package.

```java
//Get the My Domain login hostname
String myDomainHostname = DomainCreator.getOrgMyDomainHostname();

//Get the Visualforce hostname
String vfHostname = DomainCreator.getVisualforceHostname('uat');
```

In this case, in a production org with a My Domain name of **mycompany**, `myDomainHostname` returns **mycompany.my.salesforce.com**. And in the same production org with enhanced domains, `vfHostname` returns **mycompany--uat.vf.force.com**.

This example code creates a link to a Salesforce Account record. It gets the Lightning hostname associated with this org. It then gets the Account record ID and uses concatenation to build the link URL.

```java
//Get the org’s Lightning hostname
String myLightningHostname = DomainCreator.getLightningHostname();

//Get the ID of a record Account with the name 'Acme'
Account acct = [SELECT Id FROM Account WHERE Name = 'Acme' LIMIT 1];

//Build the URL to view the account record
String fullRecordURL = 'https://' + myLightningHostname + '/lightning/r/Account/' + acct.Id + '/view';
```

IN THIS SECTION:
DomainCreator Methods

DomainCreator Methods
The following are methods for DomainCreator.

IN THIS SECTION:
getContentHostname()
Returns the hostname for content stored in the org, such as files.
getExperienceCloudSitesBuilderHostname()
Returns the hostname to access Experience Builder for the org’s Experience Cloud sites.

getExperienceCloudSitesHostname()
Returns the system-managed hostname for the org’s Experience Cloud sites, such as
ExperienceCloudSitesSubdomainName.force.com. If Digital Experiences aren’t enabled, this method throws an
InvalidParameterValueException.

getExperienceCloudSitesLivePreviewHostname()
Returns the hostname to access Experience Builder Live Preview for the org’s Experience Cloud sites.

getExperienceCloudSitesPreviewHostname()
Returns the hostname to access Experience Builder Preview for the org’s Experience Cloud sites.

getLightningContainerComponentHostname(packageName)
Returns the hostname for the org’s Lightning Container Components.

getLightningHostname()
Returns the hostname for the org’s Lightning pages.

getOrgMyDomainHostname()
Returns the hostname for the org’s My Domain login domain.

getSalesforceSitesHostname()
Returns the hostname for the org’s Salesforce Sites. If Salesforce Sites aren’t enabled, this method throws an
InvalidParameterValueException.

getSetupHostname()
This method is reserved for future use.

getVisualforceHostname(packageName)
Returns the hostname for the org’s Visualforce pages.

getContentHostname()
Returns the hostname for content stored in the org, such as files.

Signature
public static String getContentHostname()  

Return Value
Type: String

gExperienceCloudSitesBuilderHostname()
Returns the hostname to access Experience Builder for the org’s Experience Cloud sites.

Signature
public static String getExperienceCloudSitesBuilderHostname()
Return Value
Type: String

getExperienceCloudSitesHostname()
Returns the system-managed hostname for the org’s Experience Cloud sites, such as
ExperienceCloudSitesSubdomainName.force.com. If Digital Experiences aren’t enabled, this method throws an
InvalidParameterValueException.

Signature
public static String getExperienceCloudSitesHostname()

Return Value
Type: String

getExperienceCloudSitesLivePreviewHostname()
Returns the hostname to access Experience Builder Live Preview for the org’s Experience Cloud sites.

Signature
public static String getExperienceCloudSitesLivePreviewHostname()

Return Value
Type: String

getExperienceCloudSitesPreviewHostname()
Returns the hostname to access Experience Builder Preview for the org’s Experience Cloud sites.

Signature
public static String getExperienceCloudSitesPreviewHostname()

Return Value
Type: String

getLightningContainerComponentHostname(String packageName)
Returns the hostname for the org’s Lightning Container Components.

Signature
public static String getLightningContainerComponentHostname(String packageName)
Parameters

packageName
Type: String
The package name for this component.
If packageName is null, this method uses the org’s namespace prefix as the package name. Otherwise, it uses the default namespace.

Return Value
Type: String

ggetLightningHostname()
Returns the hostname for the org’s Lightning pages.

Signature
public static String getLightningHostname()

Return Value
Type: String

ggetOrgMyDomainHostname()
Returns the hostname for the org’s My Domain login domain.

Signature
public static String getOrgMyDomainHostname()

Return Value
Type: String

ggetSalesforceSitesHostname()
Returns the hostname for the org’s Salesforce Sites. If Salesforce Sites aren’t enabled, this method throws an InvalidParameterValueException.

Signature
public static String getSalesforceSitesHostname()

Return Value
Type: String

ggetSetupHostname()
This method is reserved for future use.
**getVisualforceHostname (packageName)**

Returns the hostname for the org’s Visualforce pages.

**Signature**

```java
public static String getVisualforceHostname(String packageName)
```

**Parameters**

- `packageName`  
  Type: `String`  
  The package name for this component.  
  If `packageName` is `null`, this method uses the org’s namespace prefix as the package name. Otherwise, it uses the default namespace.

**Return Value**

Type: `String`

**DomainParser Class**

Use the DomainParser class to parse a domain that Salesforce hosts for the org and extract information about the domain.

**Namespace**

- `System`

**Examples**

This example code parses the org’s Lightning domain and gets the My Domain name and domain type from the `System.Domain` object.

```java
System.Domain d = DomainParser.parser('mycompany.lightning.force.com');
String myDomainName = d.getMyDomainName();
System.DomainType domainType = d.getDomainType();
```

This example code parses a known Visualforce URL to get the domain type, the org’s My Domain name, and the package name.

```java
//Parse a known URL

//Get the domain type
System.DomainType domainType = domain.getDomainType(); // Returns VISUALFORCE_DOMAIN

//Get the org’s My Domain name
String myDomainName = domain.getMyDomainName(); // Returns mycompany

//Get the package name
String packageName = domain.getPackageName(); // Returns abcpackage
```
IN THIS SECTION:
DomainParser Methods

DomainParser Methods
The following are methods for DomainParser.

parse(hostname)
Parses a passed hostname of a domain that Salesforce hosts for the org, and returns the System.Domain.

Signature
public static System.Domain parse(String hostname)

Parameters
hostname
Type: String
The label that identifies a Salesforce host, including all subdomains but without the protocol, path, or any parameters. For example, mycompany.my.site.com or mycompany--sandbox1.sandbox.my.salesforceforce.com.
If the hostname format is invalid, it isn’t a Salesforce hosted domain, or it isn’t owned by this org, this method throws an InvalidParameterValueException.

Return Value
Type: System.Domain

parse(url)
Parses a passed uniform resource locator (URL) of a domain that Salesforce hosts for the org, and returns the System.Domain.

Signature
public static System.Domain parse(System.Url url)

Parameters
url
Type: System.Url
A uniform resource locator (URL) for a Salesforce org, including all subdomains and the protocol. For example, https://mycompany--sandbox1.sandbox.my.salesforceforce.com.


If the URL format is invalid, it isn’t a Salesforce hosted domain, or it isn’t owned by this org, this method throws an InvalidParameterValueException.

Return Value
Type: System.Domain

DomainType Enum
Specifies the domain type for a System.Domain.

Usage
Use the DomainType enum to obtain the type of a domain parsed through the System.DomainParser class.

Enum Values
The following are the values of the System.DomainType enum. These values only apply to Salesforce-hosted domains.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMS_DOMAIN</td>
<td>Content Management System (CMS) public channel domains.</td>
</tr>
<tr>
<td>CONTENT_DOMAIN</td>
<td>Domains that serve content (files) stored in Salesforce.</td>
</tr>
<tr>
<td>CUSTOMER_360_ADMIN_DOMAIN</td>
<td>Customer 360 Data Manager domains.</td>
</tr>
<tr>
<td>CUSTOMER_360_DOMAIN</td>
<td>Customer 360 Data Manager Admin domains.</td>
</tr>
<tr>
<td>EXPERIENCE_CLOUD_SITES_BUILDER_DOMAIN</td>
<td>Experience Builder for Experience Cloud sites domains.</td>
</tr>
<tr>
<td>EXPERIENCE_CLOUD_SITES_DOMAIN</td>
<td>Salesforce-hosted domains that serve Experience Cloud sites.</td>
</tr>
<tr>
<td>EXPERIENCE_CLOUD_SITES_LIVE_PREVIEW_DOMAIN</td>
<td>Experience Builder Live Preview domains.</td>
</tr>
<tr>
<td>EXPERIENCE_CLOUD_SITES_PREVIEW_DOMAIN</td>
<td>Experience Builder Preview domains.</td>
</tr>
<tr>
<td>LIGHTNING_CONTAINER_COMPONENT_DOMAIN</td>
<td>Lightning Container Component domains.</td>
</tr>
<tr>
<td>LIGHTNING_DOMAIN</td>
<td>Domains that serve Lighting pages.</td>
</tr>
<tr>
<td>ORG_MY_DOMAIN</td>
<td>My Domain login domains.</td>
</tr>
<tr>
<td>SALESFORCE_SITES_DOMAIN</td>
<td>Salesforce-hosted domains that serve Salesforce Sites.</td>
</tr>
<tr>
<td>SETUP_DOMAIN</td>
<td>Reserved for future use.</td>
</tr>
<tr>
<td>VISUALFORCE_DOMAIN</td>
<td>Domains that serve Visualforce pages.</td>
</tr>
</tbody>
</table>
Double Class

Contains methods for the Double primitive data type.

Namespace

System

Usage

For more information on Double, see Double Data Type.

Double Methods

The following are methods for Double.

IN THIS SECTION:

- format()
  Returns the String value for this Double using the locale of the context user
- intValue()
  Returns the Integer value of this Double by casting it to an Integer.
- longValue()
  Returns the Long value of this Double.
- round()
  Returns the closest Long to this Double value.
- valueOf(stringToDouble)
  Returns a Double that contains the value of the specified String. As in Java, the String is interpreted as representing a signed decimal.
- valueOf(fieldValue)
  Converts the specified object to a Double value. Use this method to convert a history tracking field value or an object that represents a Double value.

format()

Returns the String value for this Double using the locale of the context user

Signature

public String format()

Return Value

Type: String
Example

```java
Double myDouble = 1261992;
System.assertEquals('1,261,992', myDouble.format());
```

**intValue()**

Returns the Integer value of this Double by casting it to an Integer.

**Signature**

```java
public Integer intValue()
```

**Return Value**

Type: Integer

**Example**

```java
Double DD1 = Double.valueOf('3.14159');
Integer value = DD1.intValue();
System.assertEquals(value, 3);
```

**longValue()**

Returns the Long value of this Double.

**Signature**

```java
public Long longValue()
```

**Return Value**

Type: Long

**Example**

```java
Double myDouble = 421994;
Long value = myDouble.longValue();
System.assertEquals(421994, value);
```

**round()**

Returns the closest Long to this Double value.

**Signature**

```java
public Long round()
```
Return Value
Type: Long

Example
```
Double D1 = 4.5;
Long L1 = D1.round();
System.assertEquals(5, L1);

Double D2 = 4.2;
Long L2 = D2.round();
System.assertEquals(4, L2);

Double D3 = -4.7;
Long L3 = D3.round();
System.assertEquals(-5, L3);
```

**valueOf**(stringToDouble)
Returns a Double that contains the value of the specified String. As in Java, the String is interpreted as representing a signed decimal.

Signature
```
public static Double valueOf(String stringToDouble)
```

Parameters
```
stringToDouble
  Type: String
```

Return Value
Type: Double

Example
```
Double DD1 = double.valueOf('3.14159');
```

**valueOf**(fieldValue)
Converts the specified object to a Double value. Use this method to convert a history tracking field value or an object that represents a Double value.

Signature
```
public static Double valueOf(Object fieldValue)
```
Parameters

fieldValue
  Type: Object

Return Value

Type: Double

Usage

Use this method with the OldValue or NewValue fields of history sObjects, such as AccountHistory, when the field type corresponds to a Double type, like a number field.

Example

```java
List<AccountHistory> ahlist =
  [SELECT Field, OldValue, NewValue
   FROM AccountHistory];
for (AccountHistory ah : ahlist) {
  System.debug('Field: ' + ah.Field);
  if (ah.field == 'NumberOfEmployees') {
    Double oldValue =
      Double.valueOf(ah.OldValue);
    Double newValue =
      Double.valueOf(ah.NewValue);
  }
}
```

EmailMessages Class

Use the methods in the EmailMessages class to interact with emails and email threading.

Namespace

System

EmailMessages Methods

The following are static methods for EmailMessages.

IN THIS SECTION:

- `getFormattedThreadingToken(recordId)`
  - Returns an email threading token that’s formatted with the correct prefix and suffix. This token can be embedded in an outbound email body, email subject, or both the body and subject. When users reply to the email, threading tokens can be used to attach responses to a record, such as a Case record in Email-to-Case.

- `getRecordIdFromEmail(subject, textBody, htmlBody)`
  - Returns the record ID corresponding to the specified email threading token, or returns null if none is found.
getFormattedThreadingToken (recordId)

Returns an email threading token that’s formatted with the correct prefix and suffix. This token can be embedded in an outbound email body, email subject, or both the body and subject. When users reply to the email, threading tokens can be used to attach responses to a record, such as a Case record in Email-to-Case.

**Signature**

```java
public static Id getFormattedThreadingToken(Id recordId)
```

**Parameters**

- **recordId**
  - Type: `Id`
  - The record ID associated with the threading token. Only Case record IDs are supported.

**Return Value**

- Type: `String`
  - The returned value is a formatted string that includes a prefix and suffix, for example:

  ```
  thread::pp5XPgfMzNf2hZdRCrNrohc::
  ```

**Usage**

Requires Lightning threading to be enabled in Email-to-Case.

When sending emails in Apex, use the returned string to match emails to a record, such as a Case record, that’s associated with the email thread. Embed the formatted token in the body or subject of outgoing emails. To find the corresponding record ID in incoming emails, use `EmailMessages.getRecordIdFromEmail(subject, textBody, htmlBody)` on page 3067.

**Example**

In this sample, we send an email with a threading token so that the email and any responses are associated with the related case.

```java
// Get your Record ID. Here, we're using a dummy Case ID.
ID caseId = Id.valueOf('500xx000000bpkTAAQ');

// Get the formatted threading token.
String formattedToken = EmailMessages.getFormattedThreadingToken(caseId);

// Create a SingleEmailMessage object.
Messaging.SingleEmailMessage email = new Messaging.SingleEmailMessage();

// Set recipients and other fields.
email.setToAddresses(new String[] {'test@example.com'});

// Append the threading token to the email body (text or html), subject,
// or both body and subject.
email.setPlainTextBody('Test Email Notification text body' + '

' + formattedToken);
email.setHtmlBody('Test Email Notification html body' + '<br><br>' + formattedToken);
email.setSubject('Test Notification ' + '

' + formattedToken + ' ]');

// ............ more fields ............
```
// Send out the email.
Messaging.sendEmail(new Messaging.SingleEmailMessage[]{email});

getRecordIdFromEmail(subject, textBody, htmlBody)

Returns the record ID corresponding to the specified email threading token, or returns null if none is found.

Signature

```java
public static Id getRecordIdFromEmail(String subject, String textBody, String htmlBody)
```

Parameters

- `subject`
  - Type: `String`
  - The subject of the email.

- `textBody`
  - Type: `String`
  - The body of the email in text format.

- `htmlBody`
  - Type: `String`
  - The body of the email in HTML format.

Return Value

- Type: `Id`
- The record ID that corresponds to the embedded threading token.

Usage

Requires Lightning threading to be enabled in Email-to-Case.

When you send emails with threading tokens embedded in the email subject, the email body, or in both the subject and body, most email clients quote the email body and maintain the email subject in a response. This method finds a corresponding record that matches the embedded threading token in a response.

Typically this method is used in Email Services so that you can provide your own handling of inbound emails using Apex code.

Example

If you implement header-based threading in your Email Services currently, we recommend that you use Lightning threading, which combines token-based threading and header-based threading. For header-based threading to continue to work, store emails as EmailMessage records with the MessagedIdentifier field set properly. With Lightning threading, you can use threading tokens as the primary threading method and rely on header-based threading as a fallback, or vice versa.

In this example, we rely on threading tokens and use header-based threading as a fallback.

```java
global class AttachEmailMessageToCaseExample implements Messaging.InboundEmailHandler {
    global Messaging.InboundEmailResult handleInboundEmail(Messaging.inboundEmail email,
```
Messaging.InboundEnvelope env) {

    // Create an InboundEmailResult object for returning the result of the
    // Apex Email Service.
    Messaging.InboundEmailResult result = new Messaging.InboundEmailResult();

    // Try to find the Case ID using threading tokens in email attributes.
    Id caseId = EmailMessages.getRecordIdFromEmail(email.subject, email.plainTextBody, email.htmlBody);

    // If we haven't found the Case ID, try finding it using headers.
    if (caseId == null) {
        caseId = Cases.getCaseIdFromEmailHeaders(email.headers);
    }

    // If a Case isn't found, create a new Case record.
    if (caseId == null) {
        Case c = new Case(Subject = email.subject);
        insert c;
        System.debug('New Case Object: ' + c);
        caseId = c.Id;
    }

    // Process recipients
    String toAddresses;
    if (email.toAddresses != null) {
        toAddresses = String.join(email.toAddresses, '; ');
    }

    // To store an EmailMessage for threading, you need at minimum
    // the Status, the MessageIdentifier, and the ParentId fields.
    EmailMessage em = new EmailMessage(
        Status = '0',
        MessageIdentifier = email.messageId,
        ParentId = caseId,
        // Other important fields.
        FromAddress = email.fromAddress,
        FromName = email.fromName,
        ToAddress = toAddresses,
        TextBody = email.plainTextBody,
        HtmlBody = email.htmlBody,
        Subject = email.subject,
        // Parse thread-index header to remain consistent with Email-to-Case.
        ClientThreadIdentifier = getClientThreadIdentifier(email.headers)
    // Other fields you wish to add.
    );

    // Insert the new EmailMessage.
    insert em;
    System.debug('New EmailMessage Object: ' + em);

    // Set the result to true. No need to send an email back to the user
    // with an error message.
public class ApexEmailService
{
    public static result EmailServiceResult = new result();

    public static void sendEmail(String from, String to, String subject, String body)
    {
        result = new result();
        result.success = true;
        // Return the result for the Apex Email Service.
        return result;
    }

    private String getClientThreadIdentifier(List<Messaging.InboundEmail.Header> headers)
    {
        if (headers == null || headers.size() == 0) return null;
        try {
            for (Messaging.InboundEmail.Header header : headers) {
                if (header.name.equalsIgnoreCase('thread-index')) {
                    Blob threadIndex = EncodingUtil.base64Decode(header.value.trim());
                    return EncodingUtil.convertToHex(threadIndex).substring(0, 44).toUpperCase();
                }
            }
        }
        catch (Exception e){
            return null;
        }
        return null;
    }
}

EncodingUtil Class

Use the methods in the EncodingUtil class to encode and decode URL strings, and convert strings to hexadecimal format.

Namespace

System

Usage

Note: You cannot use the EncodingUtil methods to move documents with non-ASCII characters to Salesforce. You can, however, download a document from Salesforce. To do so, query the ID of the document using the API query call, then request it by ID.

EncodingUtil Methods

The following are methods for EncodingUtil. All methods are static.

IN THIS SECTION:

    base64Decode(inputString)
    Converts a Base64-encoded String to a Blob representing its normal form.

    base64Encode(inputBlob)
    Converts a Blob to an unencoded String representing its normal form.

    convertFromHex(inputString)
    Converts the specified hexadecimal (base 16) string to a Blob value and returns this Blob value.
convertToHex(inputBlob)
Returns a hexadecimal (base 16) representation of the inputBlob. This method can be used to compute the client response (for example, HA1 or HA2) for HTTP Digest Authentication (RFC2617).

urlDecode(inputString, encodingScheme)
Decodes a string in application/x-www-form-urlencoded format using a specific encoding scheme, for example “UTF-8.”

urlEncode(inputString, encodingScheme)
Encodes a string into the application/x-www-form-urlencoded format using a specific encoding scheme, for example “UTF-8.”

### base64Decode(inputString)
Converts a Base64-encoded String to a Blob representing its normal form.

**Signature**

```java
public static Blob base64Decode(String inputString)
```

**Parameters**

- `inputString`
  - Type: `String`

**Return Value**

- Type: `Blob`

### base64Encode(inputBlob)
Converts a Blob to an unencoded String representing its normal form.

**Signature**

```java
public static String base64Encode(Blob inputBlob)
```

**Parameters**

- `inputBlob`
  - Type: `Blob`

**Return Value**

- Type: `String`

### convertFromHex(inputString)
Converts the specified hexadecimal (base 16) string to a Blob value and returns this Blob value.
public static Blob convertFromHex(String inputString)

Parameters
inputString
Type: String
The hexadecimal string to convert. The string can contain only valid hexadecimal characters (0-9, a-f, A-F) and must have an even number of characters.

Return Value
Type: Blob

Usage
Each byte in the Blob is constructed from two hexadecimal characters in the input string.

The convertFromHex method throws the following exceptions.
• NullPointerException — the inputString is null.
• InvalidParameterValueException — the inputString contains invalid hexadecimal characters or doesn’t contain an even number of characters.

Example
Blob blobValue = EncodingUtil.convertFromHex('4A4B4C');
System.assertEquals('JKL', blobValue.toString());

Signature
covnertToHex(Blob inputBlob)

Returns a hexadecimal (base 16) representation of the inputBlob. This method can be used to compute the client response (for example, HA1 or HA2) for HTTP Digest Authentication (RFC2617).

Signature
covnertToHex(String inputBlob)

Parameters
inputBlob
Type: Blob

Return Value
Type: String

urlDecode(inputString, encodingScheme)
Decodes a string in application/x-www-form-urlencoded format using a specific encoding scheme, for example “UTF-8.”
Signature

```java
public static String urlDecode(String inputString, String encodingScheme)
```

Parameters

- `inputString`
  Type: `String`

- `encodingScheme`
  Type: `String`

Return Value

Type: `String`

Usage

This method uses the supplied encoding scheme to determine which characters are represented by any consecutive sequence of the form `"\%xy"`. For more information about the format, see The form-urlencoded Media Type in Hypertext Markup Language - 2.0.

`urlEncode(inputString, encodingScheme)`

Encodes a string into the `application/x-www-form-urlencoded` format using a specific encoding scheme, for example “UTF-8.”

Signature

```java
public static String urlEncode(String inputString, String encodingScheme)
```

Parameters

- `inputString`
  Type: `String`

- `encodingScheme`
  Type: `String`

Return Value

Type: `String`

Usage

This method uses the supplied encoding scheme to obtain the bytes for unsafe characters. For more information about the format, see The form-urlencoded Media Type in Hypertext Markup Language - 2.0.

Example

```java
String encoded = EncodingUtil.urlEncode(url, 'UTF-8');
```
Enum Methods

An enum is an abstract data type with values that each take on exactly one of a finite set of identifiers that you specify. Apex provides built-in enums, such as `LoggingLevel`, and you can define your own enum.

All Apex enums, whether user-defined enums or built-in enums, have these common methods:

- **values**
  
  This method returns the values of the Enum as a list of the same Enum type.

- **valueOf(string enumStr)**
  
  This method converts a specified string to an enum constant value. An exception is thrown if the input string doesn’t match an enum value.

Each Enum value has the following methods that take no arguments.

- **name**
  
  Returns the name of the Enum item as a String.

- **ordinal**
  
  Returns the position of the item, as an Integer, in the list of Enum values starting with zero.

Enum values cannot have user-defined methods added to them.

For more information about Enum, see Enums.

Example

```java
Integer i = StatusCode.DELETE_FAILED.ordinal();
String s = StatusCode.DELETE_FAILED.name();
List<StatusCode> values = StatusCode.values();
StatusCode statusCodeValue = StatusCode.valueOf('delete_failed');
```

EventBus Class

Contains methods for publishing platform events.

Namespace

System

IN THIS SECTION:

  EventBus Methods

SEE ALSO:

  Platform Events Developer Guide: Publishing Platform Events

EventBus Methods

The following are methods for EventBus. All methods are static.
IN THIS SECTION:

**getOperationId(result)**
Returns the event UUID, which identifies a published event message.

**publish(event)**
Publishes the given platform event.

**publish(events)**
Publishes the given list of platform events.

---

**getOperationId(result)**
Returns the event UUID, which identifies a published event message.

**Signature**

```java
public static String getOperationId(Object result)
```

**Parameters**

- **result**
  Type: `Object`
  The `SaveResult` that is returned by the `EventBus.publish` call.

**Return Value**

Type: `String`

---

**publish(event)**
Publishes the given platform event.

**Signature**

```java
public static Database.SaveResult publish(SObject event)
```

**Parameters**

- **event**
  Type: `SObject`
  An instance of a platform event. For example, an instance of `MyEvent__e`. You must first define your platform event object in your org.

**Return Value**

Type: `Database.SaveResult`

The result of publishing the given event. `Database.SaveResult` contains information about whether the operation was successful and the errors encountered. If the `isSuccess()` method returns `true`, the publish request is queued in Salesforce and the event message is published asynchronously. For more information, see **High-Volume Platform Event Persistence**. If `isSuccess()` returns
false, the event publish operation resulted in errors, which are returned in the `Database.Error` object. This method doesn’t throw an exception due to an unsuccessful publish operation.

`Database.SaveResult` also contains the `Id` system field. The `Id` field value isn’t included in the event message delivered to subscribers. It isn’t used to identify an event message, and isn’t always unique.

**Usage**

- The platform event message is published either immediately or after a transaction is committed, depending on the publish behavior you set in the platform event definition. For more information, see Platform Event Fields in the Platform Events Developer Guide.
- Apex governor limits apply. For events configured with the Publish After Commit behavior, each method execution is counted as one DML statement against the Apex DML statement limit. You can check limit usage using the Apex `Limits.getDMLStatements()` on page 3169 method. For events configured with the Publish Immediately behavior, each method execution is counted against a separate event publishing limit of 150 `EventBus.publish()` calls. You can check limit usage using the Apex `Limits.getPublishImmediateDML()` on page 3172 method.

**publish(events)**

Publishes the given list of platform events.

**Signature**

```java
public static List<Database.SaveResult> publish(List<SObject> events)
```

**Parameters**

- `events`
  - Type: `List<SObject>`
  - A list of platform event instances. For example, a list of `MyEvent__e` objects. You must first define your platform event object in your Salesforce org.

**Return Value**

- Type: `List<Database.SaveResult>`
  - A list of results, each corresponding to the result of publishing one event. For each event, `Database.SaveResult` contains information about whether the operation was successful and the errors encountered. If the `isSuccess()` method returns `true`, the publish request is queued in Salesforce and the event message is published asynchronously. For more information, see High-Volume Platform Event Persistence. If `isSuccess()` returns `false`, the publish operation resulted in errors, which are returned in the `Database.Error` object. `EventBus.publish()` can publish some passed-in events, even when other events can’t be published due to errors. The `EventBus.publish()` method doesn’t throw exceptions caused by an unsuccessful publish operation. It’s similar in behavior to the Apex `Database.insert` method when called with the partial success option.

`Database.SaveResult` also contains the `Id` system field. The `Id` field value isn’t included in the event message delivered to subscribers. It isn’t used to identify an event message, and isn’t always unique.

**Usage**

- The platform event message is published either immediately or after a transaction is committed, depending on the publish behavior you set in the platform event definition. For more information, see Platform Event Fields in the Platform Events Developer Guide.
Apex governor limits apply. For events configured with the Publish After Commit behavior, each method execution is counted as one DML statement against the Apex DML statement limit. You can check limit usage using the Apex

`Limits.getDMLStatements()` on page 3169 method. For events configured with the Publish Immediately behavior, each method execution is counted against a separate event publishing limit of 150 `EventBus.publish()` calls. You can check limit usage using the Apex `Limits.getPublishImmediateDML()` on page 3172 method.

### Exception Class and Built-In Exceptions

An exception denotes an error that disrupts the normal flow of code execution. You can use Apex built-in exceptions or create custom exceptions. All exceptions have common methods.

All exceptions support built-in methods for returning the error message and exception type. In addition to the standard `exception` class, there are several different types of exceptions:

The following are exceptions in the `System` namespace.

<table>
<thead>
<tr>
<th>Exception</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>AssertionException</code></td>
<td>A <code>System.assert</code> failure that halts code execution. Optionally contains the custom message specified in the last (<code>msg</code>) argument to the <code>assert()</code> method.</td>
</tr>
<tr>
<td><code>AuraException</code></td>
<td>Legacy Aura-related exception. Use <code>System.AuraHandledException</code> instead.</td>
</tr>
<tr>
<td><code>AuraHandledException</code></td>
<td>Returns a custom error message to a JavaScript controller. See Returning Errors from an Apex Server-Side Controller.</td>
</tr>
<tr>
<td><code>AsyncException</code></td>
<td>Any problem with an asynchronous operation, such as failing to enqueue an asynchronous call.</td>
</tr>
<tr>
<td><code>BigObjectException</code></td>
<td>Any problem with big object records, such as connection timeouts during attempts to access or insert big object records.</td>
</tr>
<tr>
<td><code>CalloutException</code></td>
<td>Any problem with a Web service operation, such as failing to make a callout to an external system.</td>
</tr>
<tr>
<td><code>DataWeaveScriptException</code></td>
<td>Any run-time script errors that occur within DataWeave in Apex.</td>
</tr>
<tr>
<td><code>DmlException</code></td>
<td>Any problem with a DML statement, such as an <code>insert</code> statement missing a required field on a record.</td>
</tr>
<tr>
<td><code>DuplicateMessageException</code></td>
<td>Attempt to enqueue job with duplicate queueable signature</td>
</tr>
<tr>
<td><code>EmailException</code></td>
<td>Any problem with email, such as failure to deliver. For more information, see Outbound Email.</td>
</tr>
<tr>
<td><code>ExternalObjectException</code></td>
<td>Any problem with external object records, such as connection timeouts during attempts to access the data that's stored on external systems.</td>
</tr>
<tr>
<td><code>FinalException</code></td>
<td>Any attempt to mutate a read-only collection or record such as an <code>sObject</code> in an after-update trigger, or a final variable. This exception causes execution to halt.</td>
</tr>
<tr>
<td><code>FlowException</code></td>
<td>Any problem with starting flow interviews from Apex. For example, if an active version of the flow can't be found or it can't be started from Apex.</td>
</tr>
<tr>
<td><code>HandledException</code></td>
<td>A generic handled exception.</td>
</tr>
<tr>
<td>Exception</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>IllegalArgumentException</td>
<td>An illegal argument was provided to a method call. For example, a method that requires a non-null argument throws this exception if a null value is passed into the method.</td>
</tr>
<tr>
<td>InvalidHeaderException</td>
<td>An illegal header argument was provided to an Apex REST call. For example, a call to the <code>RestResponse.addHeader(name, value)</code> method throws this exception if the header name is <code>cookie</code>.</td>
</tr>
<tr>
<td>InvalidParameterValueException</td>
<td>This exception is used with both Visualforce pages and Salesforce Functions.</td>
</tr>
<tr>
<td></td>
<td><strong>Visualforce</strong></td>
</tr>
<tr>
<td></td>
<td>The exception is thrown when an invalid parameter is supplied for a method, or any problem is encountered with a URL used with Visualforce pages. For more information on Visualforce, see the Visualforce Developer’s Guide.</td>
</tr>
<tr>
<td></td>
<td><strong>Salesforce Functions</strong></td>
</tr>
<tr>
<td></td>
<td>The exception is thrown when the <code>functionName</code> parameter to <code>Function.get()</code> doesn’t have the correct project name.function name format. For more information on Salesforce functions, see <code>Function.get()</code></td>
</tr>
<tr>
<td>LimitException</td>
<td>A governor limit has been exceeded. This exception can’t be caught.</td>
</tr>
<tr>
<td>JSONException</td>
<td>Any problem with JSON serialization and deserialization operations. For more information, see the methods of <code>System.JSON</code>, <code>System.JSONParser</code>, and <code>System.JSONGenerator</code>.</td>
</tr>
<tr>
<td>ListException</td>
<td>Any problem with a list, such as attempting to access an index that is out of bounds.</td>
</tr>
<tr>
<td>MathException</td>
<td>Any problem with a mathematical operation, such as dividing by zero.</td>
</tr>
<tr>
<td>NoAccessException</td>
<td>Any problem with unauthorized access, such as trying to access an sObject that the current user doesn’t have access to. This exception is used with Visualforce pages. For more information on Visualforce, see the Visualforce Developer’s Guide.</td>
</tr>
<tr>
<td>NoDataFoundException</td>
<td>This exception is used with both Visualforce pages and Salesforce Functions.</td>
</tr>
<tr>
<td></td>
<td><strong>Visualforce</strong></td>
</tr>
<tr>
<td></td>
<td>The exception is thrown with data that doesn’t exist, such as trying to access an sObject that has been deleted. For more information on Visualforce, see the Visualforce Developer’s Guide.</td>
</tr>
<tr>
<td></td>
<td><strong>Salesforce Functions</strong></td>
</tr>
<tr>
<td></td>
<td>The exception is thrown when the project or function name provided in the <code>functionName</code> parameter to the <code>Function.get()</code> method can’t be found. For more information on Salesforce functions, see <code>Function.get()</code></td>
</tr>
<tr>
<td>NoSuchElementException</td>
<td>This exception is thrown if you try to access items that are outside the bounds of a list. This exception is used by the <code>Iterator next</code> method. For example, if <code>iterator.hasNext() == false</code> and you call <code>iterator.next()</code>, this exception is thrown. This exception is also used by the Apex Flex Queue methods and is thrown if you attempt to access a job at an invalid position in the flex queue.</td>
</tr>
<tr>
<td>Exception</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>NullPointerException</td>
<td>Any problem with dereferencing null, such as in the following code:</td>
</tr>
<tr>
<td></td>
<td>```java</td>
</tr>
<tr>
<td></td>
<td>String s;</td>
</tr>
<tr>
<td></td>
<td>s.toLowerCase(); // Since s is null, this call causes</td>
</tr>
<tr>
<td></td>
<td>// a NullPointerException</td>
</tr>
<tr>
<td>QueryException</td>
<td>Any problem with SOQL queries, such as assigning a query that returns no</td>
</tr>
<tr>
<td></td>
<td>records or more than one record to a singleton sObject variable.</td>
</tr>
<tr>
<td>RequiredFeatureMissing</td>
<td>A Chatter feature is required for code that has been deployed to an</td>
</tr>
<tr>
<td></td>
<td>organization that doesn't have Chatter enabled.</td>
</tr>
<tr>
<td>SearchException</td>
<td>Any problem with SOSL queries executed with SOAP API search() call, for</td>
</tr>
<tr>
<td></td>
<td>example, when the searchString parameter contains fewer than two characters.</td>
</tr>
<tr>
<td></td>
<td>For more information, see the <a href="#">SOAP API Developer Guide</a>.</td>
</tr>
<tr>
<td>SecurityException</td>
<td>Any problem with static methods in the Crypto utility class. For more</td>
</tr>
<tr>
<td></td>
<td>information, see <a href="#">Crypto Class</a>.</td>
</tr>
<tr>
<td>SerializationException</td>
<td>Any problem with the serialization of data. This exception is used with</td>
</tr>
<tr>
<td></td>
<td>Visualforce pages. For more information on Visualforce, see the [Visualforce</td>
</tr>
<tr>
<td></td>
<td>Developer's Guide](#).</td>
</tr>
<tr>
<td>SObjectException</td>
<td>Any problem with sObject records, such as attempting to change a field in</td>
</tr>
<tr>
<td></td>
<td>an update statement that can only be changed during insert.</td>
</tr>
<tr>
<td>StringException</td>
<td>Any problem with Strings, such as a String that is exceeding your heap size.</td>
</tr>
<tr>
<td>TypeException</td>
<td>Any problem with type conversions, such as attempting to convert the String</td>
</tr>
<tr>
<td></td>
<td>'a' to an Integer using the valueOf method.</td>
</tr>
<tr>
<td>UnexpectedException</td>
<td>A non-recoverable internal error within Salesforce has occurred. This</td>
</tr>
<tr>
<td></td>
<td>exception causes execution to halt. If necessary, contact Salesforce</td>
</tr>
<tr>
<td></td>
<td>Customer Support for more information.</td>
</tr>
<tr>
<td>VisualforceException</td>
<td>Any problem with a Visualforce page. For more information on Visualforce,</td>
</tr>
<tr>
<td></td>
<td>see the <a href="#">Visualforce Developer's Guide</a>.</td>
</tr>
<tr>
<td>XmlException</td>
<td>Any problem with the XmlStream classes, such as failing to read or write</td>
</tr>
<tr>
<td></td>
<td>XML.</td>
</tr>
</tbody>
</table>

The following is an example using the DmlException exception:
```java
Account[] accts = new Account[] {new Account(billingcity = 'San Jose')};
try {
    insert accts;
} catch (System.DmlException e) {
    for (Integer i = 0; i < e.getNumDml(); i++) {
        // Process exception here
        System.debug(e.getDmlMessage(i));
    }
}
```

For exceptions in other namespaces, see:
- [Cache Exceptions](#)
Common Exception Methods

Exception methods are all called by and operate on an instance of an exception. This table describes all instance exception methods. All types of exceptions have these methods in common.

<table>
<thead>
<tr>
<th>Name</th>
<th>Arguments</th>
<th>Return Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>getCause</td>
<td></td>
<td>Exception</td>
<td>Returns the cause of the exception as an exception object.</td>
</tr>
<tr>
<td>getLineNumber</td>
<td>Integer</td>
<td>Integer</td>
<td>Returns the line number from where the exception was thrown.</td>
</tr>
<tr>
<td>getMessage</td>
<td>String</td>
<td>String</td>
<td>Returns the error message that displays for the user.</td>
</tr>
<tr>
<td>getStackTraceString</td>
<td></td>
<td>String</td>
<td>Returns the stack trace of a thrown exception as a string.</td>
</tr>
<tr>
<td>getTypeName</td>
<td>String</td>
<td>String</td>
<td>Returns the type of exception, such as DmlException, ListException, MathException, and so on.</td>
</tr>
<tr>
<td>initCause</td>
<td>Exception cause</td>
<td>Void</td>
<td>Sets the cause for this exception, if one hasn’t already been set.</td>
</tr>
<tr>
<td>setMessage</td>
<td>String s</td>
<td>Void</td>
<td>Sets the error message that displays for the user.</td>
</tr>
</tbody>
</table>

DMLException and EmailException Methods

In addition to the common exception methods, DMLException and EmailException have these methods:

<table>
<thead>
<tr>
<th>Name</th>
<th>Arguments</th>
<th>Return Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>getDmlFieldName i</td>
<td>Integer i</td>
<td>String []</td>
<td>Returns the names of the field or fields that caused the error described by the $i^{th}$ failed row.</td>
</tr>
<tr>
<td>getDmlFields i</td>
<td>Integer i</td>
<td>Schema.sObjectField []</td>
<td>Returns the field token or tokens for the field or fields that caused the error described by the $i^{th}$ failed row. For more information on field tokens, see Dynamic Apex.</td>
</tr>
<tr>
<td>getDmlId i</td>
<td>Integer i</td>
<td>String</td>
<td>Returns the ID of the failed record that caused the error described by the $i^{th}$ failed row.</td>
</tr>
<tr>
<td>getDmlIndex i</td>
<td>Integer i</td>
<td>Integer</td>
<td>Returns the original row position of the $i^{th}$ failed row.</td>
</tr>
<tr>
<td>getDmlMessage i</td>
<td>Integer i</td>
<td>String</td>
<td>Returns the user message for the $i^{th}$ failed row.</td>
</tr>
<tr>
<td>getDmlStatusCode i</td>
<td>Integer i</td>
<td>String</td>
<td>Deprecated. Use getDmlType instead. Returns the Apex failure code for the $i^{th}$ failed row.</td>
</tr>
</tbody>
</table>
### Description

Returns the value of the System.StatusCode enum. For example:

```java
try {
    insert new Account();
} catch (System.DmlException ex) {
    System.assertEquals(System.StatusCode.REQUIRED_FIELD_MISSING, ex.getDmlType(0));
}
```

For more information about System.StatusCode, see Enums.

### getNumDml

<table>
<thead>
<tr>
<th>Name</th>
<th>Arguments</th>
<th>Return Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>getNumDml</td>
<td>Integer</td>
<td>Integer</td>
<td>Returns the number of failed rows for DML exceptions.</td>
</tr>
</tbody>
</table>

### QueryException Method

In addition to the common exception methods, QueryException has this method.

<table>
<thead>
<tr>
<th>Name</th>
<th>Arguments</th>
<th>Return Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>getInaccessibleFields</td>
<td>Map on page 3198&lt;String,Set&lt;String&gt;&gt;</td>
<td>Returns a map in which each key is an sObjectType and its corresponding value is the set of inaccessible field names in fully qualified format (Namespace__FieldName__c). Use this method to determine the cause of the QueryException. The returned map contains data only if the method that threw the QueryException is running in user mode (as opposed to the default system mode). In this code sample, it’s assumed that the user doesn’t have field level security access to the Contact.Email and Account.Website fields.</td>
<td></td>
</tr>
</tbody>
</table>

```java
try {
    List<Account> accounts = [SELECT Website, (SELECT Email FROM Contacts) FROM Account WITH USER_MODE];
} catch (QueryException qe) {
    // Handle inaccessible fields
    Map<String, Set<String>> inaccessible = qe.getInaccessibleFields();
    Set<String> accountFields = inaccessible.get('Account');
    Set<String> contactFields = inaccessible.get('Contact');
}
```
FlexQueue Class

Contains methods that reorder batch jobs in the Apex flex queue.

Namespace

System

Usage

You can place up to 100 batch jobs in a holding status for future execution. When system resources become available, the jobs are taken from the top of the Apex flex queue and moved to the batch job queue. Up to five queued or active jobs can be processed simultaneously for each org. When a job is moved out of the flex queue for processing, its status changes from Holding to Queued. Queued jobs are executed when the system is ready to process new jobs.

Use this class's methods to reorder your Holding jobs in the flex queue. As best practice and for safe usage, a FlexQueue reorder method must be the final statement in a transaction.

Example

This example moves a job to the front of the flex queue so that it's executed immediately. The job is moved by calling the System.FlexQueue.moveJobToFront() method with the high priority job ID as the parameter.

```java
ID highPriorityJobId = Database.executeBatch(new HighPriorityBatchClass(), 200);
boolean jobMovedToFrontOfQueue = FlexQueue.moveJobToFront(highPriorityJobId);
```

IN THIS SECTION:

FlexQueue Methods

SEE ALSO:

Monitoring the Apex Flex Queue

Apex Developer Guide: Using Batch Apex

FlexQueue Methods

The following are methods for FlexQueue.

IN THIS SECTION:

moveAfterJob(jobToMoveId, jobInQueueId)

Moves the job with the ID jobToMoveId immediately after the job with the ID jobInQueueId in the flex queue. You can move jobToMoveId forward or backward in the queue. If either job isn't in the queue, it throws an element-not-found exception. Returns true if the job is moved, or false if jobToMoveId is already immediately after jobInQueueId, so no change is made.
moveBeforeJob(jobToMoveId, jobInQueueId)
Moves the job with the ID jobToMoveId immediately before the job with the ID jobInQueueId in the flex queue. You can move jobToMoveId forward or backward in the queue. If either job isn’t in the queue, it throws an element-not-found exception. Returns true if the job is moved, or false if jobToMoveId is already immediately before jobInQueueId, so no change is made.

moveJobToEnd(jobId)
Moves the specified job to the end of the flex queue, to index position (size - 1). All jobs after the job’s starting position move one spot forward. If the job isn’t in the queue, it throws an element-not-found exception. Returns true if the job is moved, or false if the job is already at the end of the queue, so no change is made.

moveJobToFront(jobId)
Moves the specified job to the front of the flex queue, to index position 0. All other jobs move back one spot. If the job isn’t in the queue, it throws an element-not-found exception. Returns true if the job is moved, or false if the job is already at the front of the queue, so no change is made.

moveAfterJob (jobToMoveId, jobInQueueId)
Moves the job with the ID jobToMoveId immediately after the job with the ID jobInQueueId in the flex queue. You can move jobToMoveId forward or backward in the queue. If either job isn’t in the queue, it throws an element-not-found exception. Returns true if the job is moved, or false if jobToMoveId is already immediately after jobInQueueId, so no change is made.

Signature
public static Boolean moveAfterJob(Id jobToMoveId, Id jobInQueueId)

Parameters

jobToMoveId
Type: Id
The ID of the job to move.

jobInQueueId
Type: Id
The ID of the job to move after.

Return Value
Type: Boolean

moveBeforeJob (jobToMoveId, jobInQueueId)
Moves the job with the ID jobToMoveId immediately before the job with the ID jobInQueueId in the flex queue. You can move jobToMoveId forward or backward in the queue. If either job isn’t in the queue, it throws an element-not-found exception. Returns true if the job is moved, or false if jobToMoveId is already immediately before jobInQueueId, so no change is made.

Signature
public static Boolean moveBeforeJob(Id jobToMoveId, Id jobInQueueId)
Parameters

jobToMoveId
Type: Id
The ID of the job to move.

jobInQueueId
Type: Id
The ID of the job to use as a reference point.

Return Value
Type: Boolean

moveJobToEnd (jobId)
Moves the specified job the end of the flex queue, to index position \( \text{size} - 1 \). All jobs after the job’s starting position move one spot forward. If the job isn’t in the queue, it throws an element-not-found exception. Returns true if the job is moved, or false if the job is already at the end of the queue, so no change is made.

Signature
public static Boolean moveJobToEnd(Id jobId)

Parameters

jobId
Type: Id
The ID of the job to move.

Return Value
Type: Boolean

moveJobToFront (jobId)
Moves the specified job to the front of the flex queue, to index position 0. All other jobs move back one spot. If the job isn’t in the queue, it throws an element-not-found exception. Returns true if the job is moved, or false if the job is already at the front of the queue, so no change is made.

Signature
public static Boolean moveJobToFront(Id jobId)

Parameters

jobId
Type: Id
The ID of the job to move.
Return Value
Type: Boolean

FeatureManagement Class
Use the methods in the System.FeatureManagement class to check and modify the values of feature parameters, and to show or hide custom objects and custom permissions in your subscribers’ orgs.

Namespace
System

Usage
For information about feature parameters, see Manage Features in Second Generation Managed Packages in the Second-Generation Managed Packaging Developer Guide, or Manage Features in First-Generation Managed Packages in the First-Generation Managed Packaging Developer Guide.

The set methods (setPackageBooleanValue, setPackageDateValue, setPackageIntegerValue) use DML operations on setup sObjects. To learn more about mixing operations in a test, see Mixed DML Operations in Test Methods.

IN THIS SECTION:
FeatureManagement Methods

FeatureManagement Methods
The following are methods for FeatureManagement.

IN THIS SECTION:
changeProtection(apiName, typeApiName, protection)
Hides or reveals custom permissions, or reveals custom objects, in your subscriber’s org.

checkPackageBooleanValue(apiName)
Checks the value__c value of the FeatureParameterBoolean__c record for a feature parameter in your subscriber’s org. You set the record’s value using setPackageBooleanValue(apiName, value).

checkPackageDateValue(apiName)
Checks the value__c value of the FeatureParameterDate__c record for a feature parameter in your subscriber’s org. You can set the record’s value using setPackageDateValue(apiName, value).

checkPackageIntegerValue(apiName)
Checks the value__c value of the FeatureParameterInteger__c record for a feature parameter in your subscriber’s org. You can set the record’s value using setPackageIntegerValue(apiName, value).

checkPermission(apiName)
Checks whether a custom permission is enabled.

setPackageBooleanValue(apiName, value)
Sets the value__c value of the FeatureParameterBoolean__c record for a subscriber-to-LMO feature parameter in your subscriber’s org. You can check the record’s value using checkPackageBooleanValue(apiName).
setPackageDateValue(apiName, value)
Sets the value__c value of the FeatureParameterDate__c record for a subscriber-to-LMO feature parameter in your subscriber's org. You can check the record's value using checkPackageDateValue(apiName).

setPackageIntegerValue(apiName, value)
Sets the value__c value of the FeatureParameterInteger__c record for a subscriber-to-LMO feature parameter in your subscriber's org. You can check the record's value using checkPackageIntegerValue(apiName).

changeProtection(apiName, typeApiName, protection)
Hides or reveals custom permissions, or reveals custom objects, in your subscriber's org.

Signature

```
public static void changeProtection(String apiName, String typeApiName, String protection)
```

Parameters

apiName
Type: String
The API name of the custom object or custom permission to show or hide—for example, 'MyCustomObject__c' or 'MyCustomPermission'.

typeApiName
Type: String
The API name of the type that you want to show or hide: 'CustomObject' or 'CustomPermission'.

protection
Type: String
To show a custom object or custom permission, 'Unprotected'.
To hide a custom permission, 'Protected'.

Return Value
Type: void

Usage

⚠️ Warning: For custom permissions, you can toggle the protected value indefinitely. However, after you've released unprotected objects to subscribers, you can't set visibility to Protected. Be sure to protect any custom objects that you want to hide before you release the first package version that contains them.

To hide custom permissions in released packages:

```
FeatureManagement.changeProtection('YourCustomPermissionName', 'CustomPermission', 'Protected');
```
To unhide custom permissions and custom objects in released packages:

```aplex
FeatureManagement.changeProtection('YourCustomPermissionName', 'CustomPermission', 'Unprotected');
FeatureManagement.changeProtection('YourCustomObjectName__c', 'CustomObject', 'Unprotected');
```

### checkPackageBooleanValue(apiName)

Checks the value__c value of the FeatureParameterBoolean__c record for a feature parameter in your subscriber’s org. You set the record’s value using setPackageBooleanValue(apiName, value).

**Signature**

```aplex
public static Boolean checkPackageBooleanValue(String apiName)
```

**Parameters**

- **apiName**
  - Type: String
  - The fullName__c value of the feature parameter whose value you want to check—for example, 'SpecialAccessAvailable'.

**Return Value**

- Type: Boolean
  - The value that’s currently assigned to the value__c field on the FeatureParameterBoolean__c record that associates the feature parameter with its related license.

### checkPackageDateValue(apiName)

Checks the value__c value of the FeatureParameterDate__c record for a feature parameter in your subscriber’s org. You can set the record’s value using setPackageDateValue(apiName, value).

**Signature**

```aplex
public static Date checkPackageDateValue(String apiName)
```

**Parameters**

- **apiName**
  - Type: String
  - The fullName__c value of the feature parameter whose value you want to check—for example, 'TrialExpirationDate'.

**Return Value**

- Type: Date
The value that’s currently assigned to the value__c field on the FeatureParameterDate__c record that associates the feature parameter with its related license.

**checkPackageIntegerValue (apiName)**

Checks the value__c value of the FeatureParameterInteger__c record for a feature parameter in your subscriber’s org. You can set the record’s value using setPackageIntegerValue(apiName, value).

**Signature**

```java
public static Integer checkPackageIntegerValue(String apiName)
```

**Parameters**

- **apiName**
  - Type: String
  - The fullName__c value of the feature parameter whose value you want to check—for example, 'NumberOfLicenses'.

**Return Value**

- Type: Integer
- The value that’s currently assigned to the value__c field on the FeatureParameterInteger__c record that associates the feature parameter with its related license.

**checkPermission (apiName)**

Checks whether a custom permission is enabled.

**Signature**

```java
public static Boolean checkPermission(String apiName)
```

**Parameters**

- **apiName**
  - Type: String
  - The API name of the custom permission to check the value of—for example, 'MyCustomPermission'.

**Return Value**

- Type: Boolean
- Shows whether the permission is enabled (true) or disabled (false).

**setPackageBooleanValue (apiName, value)**

Sets the value__c value of the FeatureParameterBoolean__c record for a subscriber-to-LMO feature parameter in your subscriber’s org. You can check the record’s value using checkPackageBooleanValue (apiName).
setPackageBooleanValue(String apiName, Boolean value)

Parameters

apiName
   Type: String
      The fullName__c value of the feature parameter whose value you want to set—for example, 'SpecialAccessAvailable'.

value
   Type: Boolean
      The value to assign to the value__c field on the FeatureParameterBoolean__c record that associates the feature parameter with its related license.

Return Value

Type: void

setPackageDateValue(String apiName, Date value)

Sets the value__c value of the FeatureParameterDate__c record for a subscriber-to-LMO feature parameter in your subscriber's org. You can check the record's value using checkPackageDateValue(apiName).

setPackageIntegerValue(String apiName, Integer value)

Sets the value__c value of the FeatureParameterInteger__c record for a subscriber-to-LMO feature parameter in your subscriber's org. You can check the record's value using checkPackageIntegerValue(apiName).
Signature

public static void setPackageIntegerValue(String apiName, Integer value)

Parameters

apiName
Type: String
The fullName__c value of the feature parameter whose value you want to set—for example, 'NumberOfLicenses'.

value
Type: Integer
The value to assign to the value__c field on the FeatureParameterInteger__c record that associates the feature parameter with its related license.

Return Value
Type: void

Formula Class

Contains the recalculateFormulas method that updates (recalculates) all formula fields on the input SObjects.

Namespace

System

Usage

Recalculate formula fields on new or queried SObjects. If all data is present on the SObjects, SOQL limits are not affected. If the data required to evaluate a formula field is missing, that data is loaded and limits are changed accordingly.

The new formula values are stored in the SObjects themselves and overwrite previous values of formula fields.

Example

Account a = new Account();
a.Name = 'Salesforce';
a.BillingCity = 'San Francisco';
List<Account> accounts = new List<Account>{a};

List<FormulaRecalcResult> results = Formula.recalculateFormulas(accounts);
System.assert(results[0].isSuccess());
// Option 1
System.debug('New value: ' + accounts[0].get('My_Formula_Field__c'));
// Option 2
System.debug('New value: ' + results[0].getSObject().get('My_Formula_Field__c'));
Formula Methods
The following are methods for Formula.

IN THIS SECTION:
recalculateFormulas(sobjects)
Updates (recalculates) all formula fields on the input SObjects.

**recalculateFormulas(sobjects)**
Updates (recalculates) all formula fields on the input SObjects.

**Signature**
```
public static List<System.FormulaRecalcResult> recalculateFormulas(List<SObject> sobjects)
```

**Parameters**

- **sobjects**
  - Type: List<SObject>
  - List of sObjects whose formula fields are to be recalculated.

**Return Value**
Type: List<FormulaRecalcResult Class>

---

**FormulaRecalcFieldError Class**
The return type of the FormulaRecalcResult.getErrors method.

**Namespace**
System

IN THIS SECTION:
FormulaRecalcFieldError Methods

**FormulaRecalcFieldError Methods**
The following are methods for FormulaRecalcFieldError.

IN THIS SECTION:
get Fehler() Returns a message describing the errors encountered during formula recalculation.
getFieldName()
Returns the name of the formula recalculation error field.

getFieldError()
Returns a message describing the errors encountered during formula recalculation.

Signature
public String getFieldName()

Return Value
Type: String

getFieldError()
Returns a message describing the errors encountered during formula recalculation.

Signature
public String getFieldName()

Return Value
Type: String

FormulaRecalcResult Class
The return type of the Formula.recalculateFormulas method.

Namespace
System

Usage
Indicates the result and status of recalculating formulas on a single sObject. Holds a reference to the sObject and a list of all the fields that were recalculated.

Example
This example assumes that you have a formula field called divide__c with formula “1 / LEN(Name).

```java
List<Account> accounts = [SELECT Name FROM Account WHERE Name='Acme'];
accounts[0].Name = '';
List<FormulaRecalcResult> results = Formula.recalculateFormulas(accounts);
FormulaRecalcResult result0 = results[0];
FormulaRecalcFieldError fieldError = result0.getErrors()[0];
```
**FormulaRecalcResult Methods**

The following are methods for `FormulaRecalcResult`.

### getErrors()

If an error occurs during formula recalculation, an array of one or more database error objects, along with error codes and descriptions, is returned.

**Signature**

```java
public List<System.FormulaRecalcFieldError> getErrors()
```

**Return Value**

Type: `List<FormulaRecalcFieldError>`

### getSObject()

Returns the sObject with formulas recalculated.

**Signature**

```java
public SObject getSObject()
```

**Return Value**

Type: `SObject`

### isSuccess()

Returns a Boolean value that is set to `true` if the formula recalculation process completed successfully; otherwise, it is set to `false`.

**Signature**

```java
public boolean isSuccess()
```
Http Class
Use the Http class to initiate an HTTP request and response.

Namespace
System

Http Methods
The following are methods for Http. All are instance methods.

IN THIS SECTION:
- send(request)
  Sends an HttpRequest and returns the response.
- toString()
  Returns a string that displays and identifies the object’s properties.

**send(request)**
Sends an HttpRequest and returns the response.

**Signature**
```
public HttpResponse send(HttpRequest request)
```

**Parameters**
requests

**Request**
Type: System.HttpRequest

**Return Value**
Type: System.HttpResponse

**toString()**
Returns a string that displays and identifies the object’s properties.
HttpCalloutMock Interface

Enables sending fake responses when testing HTTP callouts.

Namespace

System

Usage

For an implementation example, see Testing HTTP Callouts by Implementing the HttpCalloutMock Interface.

HttpCalloutMock Methods

The following are methods for HttpCalloutMock.

IN THIS SECTION:

- respond(request)

Returns an HTTP response for the given request. The implementation of this method is called by the Apex runtime to send a fake response when an HTTP callout is made after Test.setMock has been called.

**respond(request)**

Returns an HTTP response for the given request. The implementation of this method is called by the Apex runtime to send a fake response when an HTTP callout is made after Test.setMock has been called.

**Signature**

public HttpResponse respond(HttpRequest request)

**Parameters**

- request
  Type: System.HttpRequest

**Return Value**

Type: System.HttpResponse
HttpRequest Class

Use the HttpRequest class to programmatically create HTTP requests like GET, POST, PATCH, PUT, and DELETE.

Namespace

System

Usage

Use the XML classes or JSON classes to parse XML or JSON content in the body of a request created by HttpRequest.

Example

The following example illustrates how you can use an authorization header with a request and handle the response.

```java
public class AuthCallout {

    public void basicAuthCallout() {
        HttpRequest req = new HttpRequest();
        req.setEndpoint('http://www.yahoo.com');
        req.setMethod('GET');

        // Specify the required user name and password to access the endpoint
        // As well as the header and header information
        String username = 'myname';
        String password = 'mypwd';

        Blob headerValue = Blob.valueOf(username + ':' + password);
        String authorizationHeader = 'Basic ' + EncodingUtil.base64Encode(headerValue);
        req.setHeader('Authorization', authorizationHeader);

        // Create a new http object to send the request object
        // A response object is generated as a result of the request
        Http http = new Http();
        HTTPResponse res = http.send(req);
        System.debug(res.getBody());
    }
}
```

Note: You can set the endpoint as a named credential URL. A named credential URL contains the scheme `callout://`, the name of the named credential, and an optional path. For example: `callout://My_Named_Credential/some_path`. A named credential specifies the URL of a callout endpoint and its required authentication parameters in one definition. Salesforce manages all authentication for Apex callouts that specify a named credential as the callout endpoint so that your code doesn’t have to. See Named Credentials as Callout Endpoints.
Compression

To compress the data you send, use setCompressed.

```java
HttpRequest req = new HttpRequest();
req.setEndPoint('my_endpoint');
req.setCompressed(true);
req.setBody('some post body');
```

If a response comes back in compressed format, `getBody` recognizes the format, uncompresses it, and returns the uncompressed value.

SEE ALSO:
- Apex Developer Guide: JSON Support
- Apex Developer Guide: XML Support

HttpRequest Constructors

The following are constructors for `HttpRequest`.

IN THIS SECTION:
- HttpRequest()
  - Creates a new instance of the `HttpRequest` class.

HttpRequest()

Creates a new instance of the `HttpRequest` class.

Signature

```java
public HttpRequest()
```

HttpRequest Methods

The following are methods for `HttpRequest`. All are instance methods.

IN THIS SECTION:
- getBody()
  - Retrieves the body of this request.
- getBodyAsBlob()
  - Retrieves the body of this request as a Blob.
getBodyDocument()  
Retrieves the body of this request as a DOM document.

getCompressed()  
If true, the request body is compressed, false otherwise.

getEndpoint()  
Retrieves the URL for the endpoint of the external server for this request.

getHeader(key)  
Retrieves the contents of the request header.

getMethod()  
Returns the type of method used by HttpRequest.

setBody(body)  
Sets the contents of the body for this request.

setBodyAsBlob(body)  
Sets the contents of the body for this request using a Blob.

setBodyDocument(document)  
Sets the contents of the body for this request. The contents represent a DOM document.

setClientCertificate(clientCert, password)  
This method is deprecated. Use setClientCertificateName instead.

setClientCertificateName(certDevName)  
If the external service requires a client certificate for authentication, set the certificate name.

setCompressed(flag)  
If true, the data in the body is delivered to the endpoint in the gzip compressed format. If false, no compression format is used.

setEndpoint(endpoint)  
Specifies the endpoint for this request.

setHeader(key, value)  
Sets the contents of the request header.

setMethod(method)  
Sets the type of method to be used for the HTTP request.

setTimeout(timeout)  
Sets a timeout for the request between 1 and 120,000 milliseconds. The timeout is the maximum time to wait for establishing the HTTP connection. The same timeout is used for waiting for the request to start. When the request is executing, such as retrieving or posting data, the connection is kept alive until the request finishes.

toString()  
Returns a string containing the URL for the endpoint of the external server for this request and the method used, for example, Endpoint=http://YourServer, Method=POST

getBody()  
Retrieves the body of this request.
Signature
public String getBody()

Return Value
Type: String

getBodyAsBlob()
Retrieves the body of this request as a Blob.

Signature
public Blob getBodyAsBlob()

Return Value
Type: Blob

getBodyDocument()
Retrieves the body of this request as a DOM document.

Signature
public Dom.Document getBodyDocument()

Return Value
Type: Dom.Document

Example
Use this method as a shortcut for:

```java
String xml = httpRequest.getBody();
Dom.Document domDoc = new Dom.Document(xml);
```

getCompressed()
If true, the request body is compressed, false otherwise.

Signature
public Boolean getCompressed()

Return Value
Type: Boolean
getEndpoint()
Retrieves the URL for the endpoint of the external server for this request.

Signature
public String getEndpoint()

Return Value
Type: String

getHeader(key)
Retrieves the contents of the request header.

Signature
public String getHeader(String key)

Parameters
key
Type: String

Return Value
Type: String

getMethod()
Returns the type of method used by HttpRequest.

Signature
public String getMethod()

Return Value
Type: String

Usage
Examples of return values:
- DELETE
- GET
- HEAD
- PATCH
- POST
- PUT
setBody (body)
Sets the contents of the body for this request.

Signature
public Void setBody(String body)

Parameters
body
Type: String

Return Value
Type: Void

Usage
Limit: 6 MB for synchronous Apex or 12 MB for asynchronous Apex.
The HTTP request and response sizes count towards the total heap size.

setBodyAsBlob (body)
Sets the contents of the body for this request using a Blob.

Signature
public Void setBodyAsBlob(Blob body)

Parameters
body
Type: Blob

Return Value
Type: Void

Usage
Limit: 6 MB for synchronous Apex or 12 MB for asynchronous Apex.
The HTTP request and response sizes count towards the total heap size.

setBodyDocument (document)
Sets the contents of the body for this request. The contents represent a DOM document.
null

null

null
Parameters

certDevName
Type: String

Return Value
Type: Void

Usage
See Using Certificates with HTTP Requests.

setCompressed(flag)
If true, the data in the body is delivered to the endpoint in the gzip compressed format. If false, no compression format is used.

Signature

class HttpRequest

public Void setCompressed(Boolean flag)

Parameters

flag
Type: Boolean

Return Value
Type: Void

setEndpoint(endpoint)
Specifies the endpoint for this request.

Signature

class HttpRequest

public Void setEndpoint(String endpoint)

Parameters

endpoint
Type: String

Possible values for the endpoint:

- Endpoint URL

  https://my_endpoint.example.com/some_path

- Named credential URL, which contains the scheme callout, the name of the named credential, and, optionally, an appended path

  callout:My_Named_Credential/some_path
Return Value
Type: Void

SEE ALSO:
 Apex Developer Guide: Named Credentials as Callout Endpoints

**setHeader(key, value)**
Sets the contents of the request header.

**Signature**

```java
public Void setHeader(String key, String value)
```

**Parameters**

- **key**
  Type: String
- **value**
  Type: String

**Return Value**
Type: Void

**Usage**
Limit 100 KB.

**setMethod(method)**
Sets the type of method to be used for the HTTP request.

**Signature**

```java
public Void setMethod(String method)
```

**Parameters**

- **method**
  Type: String
  Possible values for the method type include:
  - DELETE
  - GET
  - HEAD
  - PATCH
  - POST
PUT

TRACE

Return Value
Type: Void

Usage
You can also use this method to set any required options.

setTimeout(timeout)
Sets a timeout for the request between 1 and 120,000 milliseconds. The timeout is the maximum time to wait for establishing the HTTP connection. The same timeout is used for waiting for the request to start. When the request is executing, such as retrieving or posting data, the connection is kept alive until the request finishes.

Signature
public Void setTimeout(Integer timeout)

Parameters
timeout
  Type: Integer

Return Value
Type: Void

toString()
Returns a string containing the URL for the endpoint of the external server for this request and the method used, for example, Endpoint=http://YourServer, Method=POST

Signature
public String toString()

Return Value
Type: String

HttpResponse Class
Use the HttpResponse class to handle the HTTP response returned by the Http class.

Namespace
System
Usage

Use the XML classes or JSON Classes to parse XML or JSON content in the body of a response accessed by HttpResponse.

Example

In the following getXmlStreamReader example, content is retrieved via an HTTP callout, then the XML is parsed using the XmlStreamReader class.

```java
public class ReaderFromCalloutSample {
    public void getAndParse() {

        // Get the XML document from the endpoint
        Http http = new Http();
        HttpRequest req = new HttpRequest();
        req.setEndpoint(URL.getOrgDomainUrl().toExternalForm() + '/services/data');
        req.setMethod('GET');
        req.setHeader('Accept', 'application/xml');
        HttpResponse res = http.send(req);

        // Log the XML content
        System.debug(res.getBody());

        // Generate the HTTP response as an XML stream
        XmlStreamReader reader = res.getXmlStreamReader();

        // Read through the XML
        while(reader.hasNext()) {
            System.debug('Event Type:' + reader.getEventType());
            if (reader.getEventType() == XmlTag.START_ELEMENT) {
                System.debug(reader.getLocalName());
            }
            reader.next();
        }
    }
}
```

SEE ALSO:

Apex Developer Guide: JSON Support
Apex Developer Guide: XML Support

HttpResponse Methods

The following are methods for HttpResponse. All are instance methods.

IN THIS SECTION:

getBody()
Retrieves the body returned in the response.
getBodyAsBlob()
Retrieves the body returned in the response as a Blob.

getBodyDocument()
Retrieves the body returned in the response as a DOM document.

getHeader(key)
Retrieves the contents of the response header.

getHeaderKeys()
Retrieves an array of header keys returned in the response.

getStatus()
Retrieves the status message returned for the response.

getStatusCode()
Retrieves the value of the status code returned in the response.

getXmlStreamReader()
Returns an `XmlStreamReader` that parses the body of the callout response.

setBody(body)
Specifies the body returned in the response.

setBodyAsBlob(body)
Specifies the body returned in the response using a Blob.

setHeader(key, value)
Specifies the contents of the response header.

setStatus(status)
Specifies the status message returned in the response.

setStatusCode(statusCode)
Specifies the value of the status code returned in the response.

toString()
Returns the status message and status code returned in the response, for example:

得到体()  
Retrieves the body returned in the response.

**Signature**

```java
public String getBody()
```

**Return Value**

Type: String

**Usage**

Limit 6 MB for synchronous Apex or 12 MB for asynchronous Apex. The HTTP request and response sizes count towards the total heap size.
getBodyAsBlob()  
Retrieves the body returned in the response as a Blob.

Signature

```java
public Blob getBodyAsBlob()
```

Return Value

Type: Blob

Usage

Limit 6 MB for synchronous Apex or 12 MB for asynchronous Apex. The HTTP request and response sizes count towards the total heap size.

g gotBodyDocument()  
Retrieves the body returned in the response as a DOM document.

Signature

```java
public Dom.Document getBodyDocument()
```

Return Value

Type: Dom.Document

Example

Use it as a shortcut for:

```java
String xml = httpResponse.getBody();
Dom.Document domDoc = new Dom.Document(xml);
```

g getHeader(key)  
Retrieves the contents of the response header.

Signature

```java
public String getHeader(String key)
```

Parameters

```java
key
  Type: String
```

Return Value

Type: String
**getHeaderKeys()**
Retrieves an array of header keys returned in the response.

**Signature**

```java
public String[] getHeaderKeys()
```

**Return Value**
Type: String[]

**getStatus()**
Retrieves the status message returned for the response.

**Signature**

```java
public String getStatus()
```

**Return Value**
Type: String

**getStatusCode()**
Retrieves the value of the status code returned in the response.

**Signature**

```java
public Integer getStatusCode()
```

**Return Value**
Type: Integer

**getXmlStreamReader()**
Returns an XmlStreamReader that parses the body of the callout response.

**Signature**

```java
public XmlStreamReader getXmlStreamReader()
```

**Return Value**
Type: System.XmlStreamReader
Usage

Use it as a shortcut for:

```java
String xml = httpResponse.getBody();
XmlStreamReader xsr = new XmlStreamReader(xml);
```

**setBody (body)**

Specifies the body returned in the response.

**Signature**

```java
public Void setBody(String body)
```

**Parameters**

- `body`
  - Type: String

**Return Value**

Type: Void

**setBodyAsBlob (body)**

Specifies the body returned in the response using a Blob.

**Signature**

```java
public Void setBodyAsBlob(Blob body)
```

**Parameters**

- `body`
  - Type: Blob

**Return Value**

Type: Void

**setHeader (key, value)**

Specifies the contents of the response header.

**Signature**

```java
public Void setHeader(String key, String value)
```
Parameters

key
  Type: String
value
  Type: String

Return Value
Type: Void

**setStatus (status)**
Specifies the status message returned in the response.

**Signature**

```
public Void setStatus(String status)
```

Parameters

status
  Type: String

Return Value
Type: Void

**setStatusCode (statusCode)**
Specifies the value of the status code returned in the response.

**Signature**

```
public Void setStatusCode(Integer statusCode)
```

Parameters

statusCode
  Type: Integer

Return Value
Type: Void

**toString()**
Returns the status message and status code returned in the response, for example:
Apex Reference Guide

Id Class

Signature
public String toString()

Return Value
Type: String

Example
Status=OK, StatusCode=200

Id Class
Contains methods for the ID primitive data type.

Namespace
System

Example: Getting an sObject Token From an ID
This sample shows how to use the getSObjectType method to obtain an sObject token from an ID. The updateOwner method
in this sample accepts a list of IDs of the sObjects to update the ownerId field of. This list contains IDs of sObjects of the same type. The
second parameter is the new owner ID. Note that since it is a future method, it doesn’t accept sObject types as parameters; this is why
it accepts IDs of sObjects. This method gets the sObject token from the first ID in the list, then does a describe to obtain the object name
and constructs a query dynamicallly. It then queries for all sObjects and updates their owner ID fields to the new owner ID.
public class MyDynamicSolution {
@future
public static void updateOwner(List<ID> objIds, ID newOwnerId) {
// Validate input
System.assert(objIds != null);
System.assert(objIds.size() > 0);
System.assert(newOwnerId != null);
// Get the sObject token from the first ID
// (the List contains IDs of sObjects of the same type).
Schema.SObjectType token = objIds[0].getSObjectType();
// Using the token, do a describe
// and construct a query dynamically.
Schema.DescribeSObjectResult dr = token.getDescribe();
String queryString = 'SELECT ownerId FROM ' + dr.getName() +
' WHERE ';
for(ID objId : objIds) {
queryString += 'Id=\'' + objId + '\' OR ';
}
// Remove the last ' OR'
queryString = queryString.subString(0, queryString.length() - 4);
sObject[] objDBList = Database.query(queryString);

3111


System.assert(objDBList.size() > 0);

// Update the owner ID on the sObjects
for(Integer i=0;i<objDBList.size();i++) {
  objDBList[i].put('ownerId', newOwnerId);
}

Database.SaveResult[] srList = Database.update(objDBList, false);
for(Database.SaveResult sr : srList) {
  if (sr.isSuccess()) {
    System.debug('Updated owner ID successfully for ' +
    dr.getName() + ' ID ' + sr.getId());
  }
  else {
    System.debug('Updating ' + dr.getName() + ' returned the following errors.');
    for(Database.Error e : sr.getErrors()) {
      System.debug(e.getMessage());
    }
  }
}

Id Methods
The following are methods for Id.

IN THIS SECTION:
  addError(errorMsg)
  Marks a trigger record with a custom error message and prevents any DML operation from occurring.
  addError(errorMsg, escape)
  Marks a trigger record with a custom error message, specifies if the error message should be escaped, and prevents any DML operation from occurring.
  addError(exceptionError)
  Marks a trigger record with a custom error message and prevents any DML operation from occurring.
  addError(exceptionError, escape)
  Marks a trigger record with a custom error message and prevents any DML operation from occurring.
  getSObjectType()
  Returns the token for the sObject corresponding to this ID. This method is primarily used with describe information.
  to15(Id)
  Converts an 18-character Id value to a 15-character case-sensitive string.
  valueOf(toID)
  Converts the specified String into an ID and returns the ID.
  valueOf(str, restoreCasing)
  Converts the specified string into an ID and returns the ID. If restoreCasing is true, and the string represents an 18-character ID that has incorrect casing, the method returns an 18-character ID that is correctly aligned with its encoded casing.
addError(errorMsg)
Marks a trigger record with a custom error message and prevents any DML operation from occurring.

Signature
public Void addError(String errorMsg)

Parameters
errorMsg
Type: String
The error message to mark the record with.

Return Value
Type: Void

Usage
This method is similar to the addError(errorMsg) sObject method.

Note: This method escapes any HTML markup in the specified error message. The escaped characters are: \n, <, >, &", \, \u2028, \u2029, and \u00a9. As a result, HTML markup is not rendered; instead, it is displayed as text in the Salesforce user interface.

Example
Trigger.new[0].Id.addError('bad');

addError(errorMsg, escape)
Marks a trigger record with a custom error message, specifies if the error message should be escaped, and prevents any DML operation from occurring.

Signature
public Void addError(String errorMsg, Boolean escape)

Parameters
errorMsg
Type: String
The error message to mark the record with.

escape
Type: Boolean
Indicates whether any HTML markup in the custom error message should be escaped (true) or not (false). This parameter is ignored in both Lightning Experience and the Salesforce mobile app, and the HTML is always escaped. The escape parameter only applies in Salesforce Classic.
Return Value
Type: Void

Usage
The escaped characters are: \n, <, >, &", \, \u2028, \u2029, and \u00a9. As a result, HTML markup is not rendered; instead, it is displayed as text in the Salesforce user interface.

⚠️ Warning: Be cautious if you specify false for the escape argument. Unescaped strings displayed in the Salesforce user interface can represent a vulnerability in the system because these strings might contain harmful code. If you want to include HTML markup in the error message, call this method with a false escape argument. Make sure that you escape any dynamic content, such as input field values. Otherwise, specify true for the escape argument or call addError(String errorMsg) instead.

Example
Trigger.new[0].Id.addError('Fix & resubmit', false);

addError(exceptionError)
Marks a trigger record with a custom error message and prevents any DML operation from occurring.

Signature
public Void addError(Exception exceptionError)

Parameters
exceptionError
Type: System.Exception
An Exception object or a custom exception object that contains the error message to mark the record with.

Return Value
Type: Void

Usage
This method is similar to the addError(exceptionError) sObject method.
This method escapes any HTML markup in the specified error message. The escaped characters are: \n, <, >, &", \, \u2028, \u2029, and \u00a9. As a result, HTML markup is not rendered; instead, it is displayed as text in the Salesforce user interface.

Example
public class MyException extends Exception{}
Trigger.new[0].Id.addError(new MyException('Invalid Id'));
addError(exceptionError, escape)
 Marks a trigger record with a custom error message and prevents any DML operation from occurring.

Signature

```java
class MyException extends Exception{}
```n
account a = new account();
a.addError(new MyException('Invalid Id & other issues'), false);

getSObjectType()

Returns the token for the sObject corresponding to this ID. This method is primarily used with describe information.

Signature

```java
public Schema.SObjectType getSObjectType()
```
Return Value
Type: Schema.SObjectType

Usage
For more information about describes, see Understanding Apex Describe Information.

Example
```java
account a = new account(name = 'account');
insert a;
Id myId = a.id;
system.assertEquals(Schema.Account.SObjectType, myId.getSobjectType());
```

tol5(Id)
Converts an 18-character Id value to a 15-character case-sensitive string.

Signature
```java
public static string to15()
```

Return Value
Type: String

Example
```java
String Id_15_char = '0D5B000001DVM9t';
String Id_18_char = '0D5B000001DVM9tkAh';
ID testId = Id_18_char;
System.assertEquals(testId.to15(),Id_15_char);
```

valueOf(toID)
Converts the specified String into an ID and returns the ID.

Signature
```java
public static ID valueOf(String toID)
```

Parameters
```java
toID
  Type: String
```

Return Value
Type: ID
Example

Id myId = Id.valueOf('001xa00003D1lo');

Versioned Behavior Changes
In API version 54.0 and later, assignment of an invalid 15 or 18 character ID to a variable results in a System.StringException exception.

valueOf(str, restoreCasing)
Converts the specified string into an ID and returns the ID. If restoreCasing is true, and the string represents an 18-character ID that has incorrect casing, the method returns an 18-character ID that is correctly aligned with its encoded casing.

Signature
public static Id valueOf(String str, Boolean restoreCasing)

Parameters
str
Type: String
String to be converted to an ID
restoreCasing
Type: Boolean
If set to true, and str represents an 18-character ID that has incorrect casing, the method returns an 18-character ID that is correctly aligned with its casing.

Return Value
Type: Id
The return value depends on both the str and the restoreCasing parameter values.

⚠️ Note: If the str is invalid, the method throws a System.StringException exception.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>restoreCasing=true</th>
<th>restoreCasing=false</th>
</tr>
</thead>
<tbody>
<tr>
<td>Valid 15-character str value</td>
<td>Because the 15-character input value can't be encoded for casing, the method throws a System.StringException.</td>
<td>The method returns a 15-character ID.</td>
</tr>
<tr>
<td>Valid 18-character str value</td>
<td>The method returns an 18-character ID that is correctly aligned with its encoded casing.</td>
<td>The method doesn't attempt to correctly align the casing of the 18-character ID and returns an 18-character ID.</td>
</tr>
</tbody>
</table>

Ideas Class
Represents zone ideas.
Namespace

System

Usage

Ideas is a zone of users who post, vote for, and comment on ideas. An Ideas zone provides an online, transparent way for you to attract, manage, and showcase innovation.

A set of recent replies (returned by methods, see below) includes ideas that a user posted or commented on that already have comments posted by another user. The returned ideas are listed based on the time of the last comment made by another user, with the most recent ideas appearing first.

The userID argument is a required argument that filters the results so only the ideas that the specified user has posted or commented on are returned.

The communityID argument filters the results so only the ideas within the specified zone are returned. If this argument is the empty string, then all recent replies for the specified user are returned regardless of the zone.

For more information on ideas, see “Using Ideas” in the Salesforce online help.

Example

The following example finds ideas in a specific zone that have similar titles as a new idea:

```java
public class FindSimilarIdeasController {

    public static void test() {
        // Instantiate a new idea
        Idea idea = new Idea();

        // Specify a title for the new idea
        idea.Title = 'Increase Vacation Time for Employees';

        // Specify the communityID (INTERNAL_IDEAS) in which to find similar ideas.
        Community community = [ SELECT Id FROM Community WHERE Name = 'INTERNAL_IDEAS' ];

        idea.CommunityId = community.Id;

        ID[] results = Ideas.findSimilar(idea);
    }
}
```

The following example uses a Visualforce page in conjunction with a custom controller, that is, a special Apex class. For more information on Visualforce, see the Visualforce Developer’s Guide.

This example creates an Apex method in the controller that returns unread recent replies. You can leverage this same example for the getAllRecentReplies and getReadRecentReplies methods. For this example to work, there must be ideas posted to the zone. In addition, at least one zone member must have posted a comment to another zone member’s idea or comment.

```java
// Create an Apex method to retrieve the recent replies marked as unread in all communities
public Idea[] getUnreadRecentReplies() {
    Idea[] recentReplies;
    if (recentReplies == null) {
```
Id[] recentRepliesIds = Ideas.getUnreadRecentReplies(UserInfo.getUserId(), '');

recentReplies = [SELECT Id, Title FROM Idea WHERE Id IN :recentRepliesIds];

return recentReplies;

The following is the markup for a Visualforce page that uses the above custom controller to list unread recent replies.

```
<apex:page controller="IdeasController" showHeader="false">
    <apex:dataList value="{!unreadRecentReplies}" var="recentReplyIdea">
        <a href="/apex/viewIdea?id={!recentReplyIdea.Id}"
            <apex:outputText value="{!recentReplyIdea.Title}" escape="true"/>
    </apex:dataList>
</apex:page>
```

The following example uses a Visualforce page in conjunction with a custom controller to list ideas. Then, a second Visualforce page and custom controller is used to display a specific idea and mark it as read. For this example to work, there must be ideas posted to the zone.

```
// Create a controller to use on a VisualForce page to list ideas
public class IdeaListController {
    public final Idea[] ideas {get; private set;}

    public IdeaListController() {
        Integer i = 0;
        ideas = new Idea[10];
        for (Idea tmp : Database.query
            ('SELECT Id, Title FROM Idea WHERE Id != null AND parentIdeaId = null LIMIT 10')) {
            i++;
            ideas.add(tmp);
        }
    }
}
```

The following is the markup for a Visualforce page that uses the above custom controller to list ideas:

```
<apex:page controller="IdeaListController" tabStyle="Idea" showHeader="false">
    <apex:dataList value="{!ideas}" var="idea" id="ideaList">
        <a href="/apex/viewIdea?id={!idea.id}"
            <apex:outputText value="{!idea.title}" escape="true"/>
    </apex:dataList>
</apex:page>
```

The following example also uses a Visualforce page and custom controller, this time, to display the idea that is selected on the above idea list page. In this example, the markRead method marks the selected idea and associated comments as read by the user that is currently logged in. Note that the markRead method is in the constructor so that the idea is marked read immediately when the user

3119
goes to a page that uses this controller. For this example to work, there must be ideas posted to the zone. In addition, at least one zone member must have posted a comment to another zone member's idea or comment.

```java
// Create an Apex method in the controller that marks all comments as read for the
// selected idea
public class ViewIdeaController {
    private final String id = System.currentPage().getParameters().get('id');

    public ViewIdeaController(ApexPages.StandardController controller) {
        Ideas.markRead(id);
    }
}
```

The following is the markup for a Visualforce page that uses the above custom controller to display the idea as read.

```xml
<apex:page standardController="Idea" extensions="ViewIdeaController" showHeader="false">
    <h2><apex:outputText value="{!idea.title}" /></h2>
    <apex:outputText value="{!idea.body}" />
</apex:page>
```

### Ideas Methods

The following are methods for Ideas. All methods are static.

**IN THIS SECTION:**
- **findSimilar(idea)**
  - Returns a list of similar ideas based on the title of the specified idea.
- **getAllRecentReplies(userID, communityID)**
  - Returns ideas that have recent replies for the specified user or zone. This includes all read and unread replies.
- **getReadRecentReplies(userID, communityID)**
  - Returns ideas that have recent replies marked as read.
- **getUnreadRecentReplies(userID, communityID)**
  - Returns ideas that have recent replies marked as unread.
- **markRead(ideaID)**
  - Marks all comments as read for the user that is currently logged in.

#### findSimilar(idea)

Returns a list of similar ideas based on the title of the specified idea.

**Signature**

```java
public static ID[] findSimilar(Idea idea)
```
Parameters

idea
    Type: Idea

Return Value

Type: ID[]

Usage

Each findSimilar call counts against the SOSL query limits. See Execution Governors and Limits.

getAllRecentReplies(userID, communityID)

Returns ideas that have recent replies for the specified user or zone. This includes all read and unread replies.

Signature

public static ID[] getAllRecentReplies(String userID, String communityID)

Parameters

userID
    Type: String

communityID
    Type: String

Return Value

Type: ID[]

Usage

Each getAllRecentReplies call counts against the SOQL query limits. See Execution Governors and Limits.

getReadRecentReplies(userID, communityID)

Returns ideas that have recent replies marked as read.

Signature

public static ID[] getReadRecentReplies(String userID, String communityID)

Parameters

userID
    Type: String

communityID
    Type: String
Return Value
Type: ID[]

Usage
Each `getReadRecentReplies` call counts against the SOQL query limits. See Execution Governors and Limits.

`getUnreadRecentReplies(userID, communityID)`
Returns ideas that have recent replies marked as unread.

Signature
```java
public static ID[] getUnreadRecentReplies(String userID, String communityID)
```

Parameters
- `userID`
  Type: String
- `communityID`
  Type: String

Return Value
Type: ID[]

Usage
Each `getUnreadRecentReplies` call counts against the SOQL query limits. See Execution Governors and Limits.

`markRead(ideaID)`
Marks all comments as read for the user that is currently logged in.

Signature
```java
public static Void markRead(String ideaID)
```

Parameters
- `ideaID`
  Type: String

Return Value
Type: Void

InstallHandler Interface
Enables custom code to run after a managed package installation or upgrade.
App developers can implement this interface to specify Apex code that runs automatically after a subscriber installs or upgrades a managed package. The package install or upgrade can be customized based on details of the subscriber's organization. For instance, you can use the script to populate custom settings, create sample data, send an email to the installer, notify an external system, or kick off a batch operation to populate a new field across a large set of data.

The post install script is invoked after tests have been run, and is subject to default governor limits. It runs as a special system user that represents your package, so all operations performed by the script appear to be done by your package. You can access this user by using UserInfo. You only see this user at runtime, not while running tests.

If the script fails, the install or upgrade is aborted. Any errors in the script are emailed to the user specified in the Notify on Apex Error field of the package. If no user is specified, the install or upgrade details are unavailable.

The post install script has the following additional properties.

- It can initiate batch, scheduled, and future jobs.
- It can’t access Session IDs.
- It can only perform callouts using an async operation. The callout occurs after the script is run and the install is complete and committed.
- It can’t call another Apex class in the package if that Apex class uses the with sharing keyword. This keyword can prevent the package from successfully installing. See the Apex Developer Guide to learn more.

The InstallHandler interface has a single method called onInstall, which specifies the actions to be performed on install or upgrade.

```java
public interface InstallHandler {
    void onInstall(InstallContext context);
}
```

The onInstall method takes a context object as its argument, which provides the following information.

- The org ID of the organization in which the installation takes place.
- The user ID of the user who initiated the installation.
- The version number of the previously installed package (specified using the Version class). The version is always a three-part number, such as 1.2.0.
- Whether the installation is an upgrade.
- Whether the installation is a push.

The context argument is an object whose type is the InstallContext interface. This interface is automatically implemented by the system. The following definition of the InstallContext interface shows the methods you can call on the context argument.

```java
public interface InstallContext {
    ID organizationId();
    ID installerId();
    Boolean isUpgrade();
    Boolean isPush();
    Version previousVersion();
}
```
InstallHandler Methods

The following are methods for InstallHandler.

onInstall(context)
Specifies the actions to be performed on install/upgrade.

Signature
public Void onInstall(InstallContext context)

Parameters
context
Type: System.InstallContext

Return Value
Type: Void

InstallHandler Example Implementation

The following sample post install script performs these actions on package install/upgrade.

• If the previous version is null, that is, the package is being installed for the first time, the script:
  – Creates a new Account called Newco and verifies that it was created.
  – Creates a new instance of the custom object Survey, called Client Satisfaction Survey.
  – Sends an email message to the subscriber confirming installation of the package.

• If the previous version is 1.0, the script creates a new instance of Survey called "Upgrading from Version 1.0".

• If the package is an upgrade, the script creates a new instance of Survey called "Sample Survey during Upgrade".

• If the upgrade is being pushed, the script creates a new instance of Survey called "Sample Survey during Push".

```java
public class PostInstallClass implements InstallHandler {
    global void onInstall(InstallContext context) {
        if(context.previousVersion() == null) {
            Account a = new Account(name='Newco');
            insert(a);
        }
    }
}
```
You can test a post install script using the new testInstall method of the Test class. This method takes the following arguments.

- A class that implements the InstallHandler interface.
- A Version object that specifies the version number of the existing package.
- An optional Boolean value that is true if the installation is a push. The default is false.

This sample shows how to test a post install script implemented in the PostInstallClass Apex class.

```apex
@isTest
static void testInstallScript() {
    PostInstallClass postinstall = new PostInstallClass();
    Test.testInstall(postinstall, null);
    Test.testInstall(postinstall, new Version(1,0), true);
    List<Account> a = [Select id, name from Account where name ='Newco'];
    System.assertEquals(a.size(), 1, 'Account not found');
}
```

**Integer Class**

Contains methods for the Integer primitive data type.
Namespace

System

Usage

For more information on integers, see Integer Data Type.

Integer Methods

The following are methods for Integer.

IN THIS SECTION:

format()
Returns the integer as a string using the locale of the context user.

valueOf(stringToInteger)
Returns an Integer that contains the value of the specified String. As in Java, the String is interpreted as representing a signed decimal integer.

valueOf(fieldValue)
Converts the specified object to an Integer. Use this method to convert a history tracking field value or an object that represents an Integer value.

format()
Returns the integer as a string using the locale of the context user.

Signature

public String format()

Return Value

Type: String

Example

integer myInt = 22;
system.assertEquals('22', myInt.format());

valueOf(stringToInteger)
Returns an Integer that contains the value of the specified String. As in Java, the String is interpreted as representing a signed decimal integer.

Signature

public static Integer valueOf(String stringToInteger)
Parameters

`stringToInteger`
Type: `String`

Return Value
Type: `Integer`

Example
```
Integer myInt = Integer.valueOf('123');
```

`valueOf(fieldValue)`
Converts the specified object to an Integer. Use this method to convert a history tracking field value or an object that represents an Integer value.

Signature
```
public static Integer valueOf(Object fieldValue)
```

Parameters

`fieldValue`
Type: `Object`

Return Value
Type: `Integer`

Usage
Use this method with the `OldValue` or `NewValue` fields of history sObjects, such as `AccountHistory`, when the field type corresponds to an Integer type, like a number field.

Example:
```
List<AccountHistory> ahlist =
    [SELECT Field, OldValue, NewValue
     FROM AccountHistory];
for(AccountHistory ah : ahlist) {
    System.debug('Field: ' + ah.Field);
    if (ah.field == 'NumberOfEmployees') {
        Integer oldValue =
            Integer.valueOf(ah.OldValue);
        Integer newValue =
            Integer.valueOf(ah.NewValue);
    }
}
JSON Class

Contains methods for serializing Apex objects into JSON format and deserializing JSON content that was serialized using the `serialize` method in this class.

Namespace

`System`

Usage

Use the methods in the `System.JSON` class to perform round-trip JSON serialization and deserialization of Apex objects.

SEE ALSO:

* Apex Developer Guide: Roundtrip Serialization and Deserialization*

JSON Methods

The following are methods for JSON. All methods are static.

IN THIS SECTION:

- `createGenerator(prettyPrint)`
  Returns a new JSON generator.
- `createParser(jsonString)`
  Returns a new JSON parser.
- `deserialize(jsonString, apexType)`
  Deserializes the specified JSON string into an Apex object of the specified type.
- `deserializeStrict(jsonString, apexType)`
  Deserializes the specified JSON string into an Apex object of the specified type.
- `deserializeUntyped(jsonString)`
  Deserializes the specified JSON string into collections of primitive data types.
- `serialize(objectToSerialize)`
  Serializes Apex objects into JSON content.
- `serialize(objectToSerialize, suppressApexObjectNulls)`
  Suppresses `null` values when serializing Apex objects into JSON content.
- `serializePretty(objectToSerialize)`
  Serializes Apex objects into JSON content and generates indented content using the pretty-print format.
- `serializePretty(objectToSerialize, suppressApexObjectNulls)`
  Suppresses `null` values when serializing Apex objects into JSON content and generates indented content using the pretty-print format.

**createGenerator (prettyPrint)**

Returns a new JSON generator.
JSON Class

**Signature**

```java
public static System.JSONGenerator createGenerator(Boolean prettyPrint)
```

**Parameters**

- `prettyPrint`
  - **Type**: `Boolean`
  - Determines whether the JSON generator creates JSON content in pretty-print format with the content indented. Set to `true` to create indented content.

**Return Value**

- **Type**: `System.JSONGenerator`

---

**createParser(jsonString)**

Returns a new JSON parser.

**Signature**

```java
public static System.JSONParser createParser(String jsonString)
```

**Parameters**

- `jsonString`
  - **Type**: `String`
  - The JSON content to parse.

**Return Value**

- **Type**: `System.JSONParser`

---

**deserialize(jsonString, apexType)**

Deserializes the specified JSON string into an Apex object of the specified type.

**Signature**

```java
public static Object deserialize(String jsonString, System.Type apexType)
```

**Parameters**

- `jsonString`
  - **Type**: `String`
  - The JSON content to deserialize.

- `apexType`
  - **Type**: `System.Type`
  - The Apex type of the object that this method creates after deserializing the JSON content.
Return Value
Type: Object

Usage
If the JSON content contains attributes not present in the System.Type argument, such as a missing field or object, deserialization fails in some circumstances. When deserializing JSON content into a custom object or an sObject using Salesforce API version 34.0 or earlier, this method throws a runtime exception when passed extraneous attributes. When deserializing JSON content into an Apex class in any API version, or into an object in API version 35.0 or later, no exception is thrown. When no exception is thrown, this method ignores extraneous attributes and parses the rest of the JSON content.

Example
The following example deserializes a Decimal value.

```java
Decimal n = (Decimal)JSON.deserialize('100.1', Decimal.class);
System.assertEquals(n, 100.1);
```

`deserializeStrict(jsonString, apexType)`
Deserializes the specified JSON string into an Apex object of the specified type.

Signature
```
public static Object deserializeStrict(String jsonString, System.Type apexType)
```

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>jsonString</code></td>
<td>String</td>
<td>The JSON content to deserialize.</td>
</tr>
<tr>
<td><code>apexType</code></td>
<td>System.Type</td>
<td>The Apex type of the object that this method creates after deserializing the JSON content.</td>
</tr>
</tbody>
</table>

Return Value
Type: Object

Usage
All attributes in the JSON string must be present in the specified type. If the JSON content contains attributes not present in the System.Type argument, such as a missing field or object, deserialization fails in some circumstances. When deserializing JSON content with extraneous attributes into an Apex class, this method throws an exception in all API versions. However, no exception is thrown when you use this method to deserialize JSON content into a custom object or an sObject.
Example
The following example deserializes a JSON string into an object of a user-defined type represented by the `Car` class, which this example also defines.

```java
class Car {
    public String make;
    public String year;
}

public void parse() {
    Car c = JSON.deserializeStrict(
        '{"make":"SFDC","year":"2020"}',
        Car.class);
    System.assertEquals(c.make, 'SFDC');
    System.assertEquals(c.year, '2020');
}
```

deserializeUntyped(String jsonString)
Deserializes the specified JSON string into collections of primitive data types.

**Signature**

```java
public static Object deserializeUntyped(String jsonString)
```

**Parameters**

`jsonString`
Type: `String`
The JSON content to deserialize.

**Return Value**
Type: Object

Example
The following example deserializes a JSON representation of an appliance object into a map that contains primitive data types and further collections of primitive types. It then verifies the deserialized values.

```java
String jsonInput = '{
    "description":"An appliance",
    "accessories": [{"right":"door handle1", "left":"door handle2" }],
    "dimensions": {
        "height": 5.5,
        "width": 3.0,
        "depth": 2.2
    },
    "type": null,
    "inventory": 2000,
    "price": 1023.45
}'
```
Map<String, Object> m =
  (Map<String, Object>)
  JSON.deserializeUntyped(jsonInput);

System.assertEquals(
  'An appliance', m.get('description'));

List<Object> a =
  (List<Object>)m.get('accessories');
System.assertEquals('powerCord', a[0]);
Map<String, Object> a2 =
  (Map<String, Object>)a[1];
System.assertEquals(
  'door handle1', a2.get('right'));
System.assertEquals(
  'door handle2', a2.get('left'));

Map<String, Object> dim =
  (Map<String, Object>)m.get('dimensions');
System.assertEquals(5.5, dim.get('height'));
System.assertEquals(3.0, dim.get('width'));
System.assertEquals(2.2, dim.get('depth'));

System.assertEquals(null, m.get('type'));
System.assertEquals(2000, m.get('inventory'));
System.assertEquals(1023.45, m.get('price'));
System.assertEquals(true, m.get('isShipped'));
System.assertEquals('123', m.get('modelNumber'));

**serialize(objectToSerialize)**

Serializes Apex objects into JSON content.

**Signature**

```
public static String serialize(Object objectToSerialize)
```

**Parameters**

**objectToSerialize**

Type: Object

The Apex object to serialize.
Return Value
Type: String

Example
The following example serializes a new DateTime value.

```java
Datetime dt = Datetime.newInstance(
    Date.newInstance(
        2011, 3, 22),
    Time.newInstance(
        1, 15, 18, 0));
String str = JSON.serialize(dt);
System.assertEquals(
    "2011-03-22T08:15:18.000Z",
    str);
```

`serialize(objectToSerialize, suppressApexObjectNulls)`
Suppresses null values when serializing Apex objects into JSON content.

Signature
```
public static String serialize(Object objectToSerialize, Boolean suppressApexObjectNulls)
```

Parameters
- `objectToSerialize`  
  Type: Object  
  The Apex object to serialize.
- `suppressApexObjectNulls`  
  Type: Boolean  
  If true, remove null values before serializing the JSON object.

Note: This parameter doesn’t apply to sObjects retrieved via SOQL.

Return Value
Type: String

Usage
This method allows you to specify whether to suppress null values when serializing Apex objects into JSON content.

`serializePretty(objectToSerialize)`
Serializes Apex objects into JSON content and generates indented content using the pretty-print format.
**Signature**

```java
public static String serializePretty(Object objectToSerialize)
```

**Parameters**

- `objectToSerialize`  
  Type: Object  
  The Apex object to serialize.

**Return Value**

Type: String

**serializePretty(objectToSerialize, suppressApexObjectNulls)**

Suppresses null values when serializing Apex objects into JSON content and generates indented content using the pretty-print format.

**Signature**

```java
public static String serializePretty(Object objectToSerialize, Boolean suppressApexObjectNulls)
```

**Parameters**

- `objectToSerialize`  
  Type: Object  
  The Apex object to serialize.

- `suppressApexObjectNulls`  
  Type: Boolean  
  If true, remove null values before serializing the JSON object.

**Note:** This parameter doesn't apply to sObjects retrieved via SOQL.

**Return Value**

Type: String

**JSONGenerator Class**

Contains methods used to serialize objects into JSON content using the standard JSON encoding.

**Namespace**

`System`
Usage
The `System.JSONGenerator` class is provided to enable the generation of standard JSON-encoded content and gives you more control on the structure of the JSON output.

SEE ALSO:
   * Apex Developer Guide: JSON Generator

**JSONGenerator Methods**

The following are methods for `JSONGenerator`. All are instance methods.

**IN THIS SECTION:**
- `close()`: Closes the JSON generator.
- `getAsString()`: Returns the generated JSON content.
- `isClosed()`: Returns `true` if the JSON generator is closed; otherwise, returns `false`.
- `writeBlob(blobValue)`: Writes the specified `Blob` value as a base64-encoded string.
- `write BlobField(fieldName, blobValue)`: Writes a field name and value pair using the specified field name and BLOB value.
- `writeBoolean(blobValue)`: Writes the specified Boolean value.
- `writeBooleanField(fieldName, booleanValue)`: Writes a field name and value pair using the specified field name and Boolean value.
- `writeDate(dateValue)`: Writes the specified date value in the ISO-8601 format.
- `writeDateField(fieldName, dateValue)`: Writes a field name and value pair using the specified field name and date value. The date value is written in the ISO-8601 format.
- `writeDateTime(datetimeValue)`: Writes the specified date and time value in the ISO-8601 format.
- `writeDateTimeField(fieldName, datetimeValue)`: Writes a field name and value pair using the specified field name and date and time value. The date and time value is written in the ISO-8601 format.
- `writeEndArray()`: Writes the ending marker of a JSON array (`'['`).
- `writeEndObject()`: Writes the ending marker of a JSON object (`']`).
- `writeFieldName(fieldName)`: Writes a field name.
writeId(identifier)
Writes the specified ID value.

writeIdField(fieldName, identifier)
Writes a field name and value pair using the specified field name and identifier value.

writeNull()
Writes the JSON null literal value.

writeNullField(fieldName)
Writes a field name and value pair using the specified field name and the JSON null literal value.

writeNumber(number)
Writes the specified decimal value.

writeNumber(number)
Writes the specified double value.

writeNumber(number)
Writes the specified integer value.

writeNumber(number)
Writes the specified long value.

writeNumberField(fieldName, number)
Writes a field name and value pair using the specified field name and decimal value.

writeNumberField(fieldName, number)
Writes a field name and value pair using the specified field name and double value.

writeNumberField(fieldName, number)
Writes a field name and value pair using the specified field name and integer value.

writeNumberField(fieldName, number)
Writes a field name and value pair using the specified field name and long value.

writeObject(anyObject)
Writes the specified Apex object in JSON format.

writeObjectField(fieldName, value)
Writes a field name and value pair using the specified field name and Apex object.

writeStartArray()
Writes the starting marker of a JSON array ('[').

writeStartObject()
Writes the starting marker of a JSON object ('{').

writeString(stringValue)
Writes the specified string value.

writeStringField(fieldName, stringValue)
Writes a field name and value pair using the specified field name and string value.

writeTime(timeValue)
Writes the specified time value in the ISO-8601 format.

writeTimeField(fieldName, timeValue)
Writes a field name and value pair using the specified field name and time value in the ISO-8601 format.
**close()**  
Closes the JSON generator.

**Signature**  
```java
public Void close()
```  

**Return Value**  
Type: Void

**Usage**  
No more content can be written after the JSON generator is closed.

**getAsString()**  
Returns the generated JSON content.

**Signature**  
```java
public String getAsString()
```  

**Return Value**  
Type: String

**Usage**  
This method closes the JSON generator if it isn't closed already.

**isClosed()**  
Returns **true** if the JSON generator is closed; otherwise, returns **false**.

**Signature**  
```java
public Boolean isClosed()
```  

**Return Value**  
Type: Boolean

**writeBlob(blobValue)**  
Writes the specified **Blob** value as a base64-encoded string.

**Signature**  
```java
public Void writeBlob(Blob blobValue)
```
Parameters
blobValue
Type: Blob

Return Value
Type: Void

writeBlobField(fieldName, blobValue)
Writes a field name and value pair using the specified field name and BLOB value.

Signature
public Void writeBlobField(String fieldName, Blob blobValue)

Parameters
fieldName
Type: String
blobValue
Type: Blob

Return Value
Type: Void

writeBoolean(blobValue)
Writes the specified Boolean value.

Signature
public Void writeBoolean(Boolean blobValue)

Parameters
blobValue
Type: Boolean

Return Value
Type: Void

writeBooleanField(fieldName, booleanValue)
Writes a field name and value pair using the specified field name and Boolean value.
**Signature**

```java
public Void writeBooleanField(String fieldName, Boolean booleanValue)
```

**Parameters**

- **fieldName**
  - Type: `String`

- **booleanValue**
  - Type: `Boolean`

**Return Value**

Type: `Void`

**writeDate(dateValue)**

Writes the specified date value in the ISO-8601 format.

**Signature**

```java
public Void writeDate(Date dateValue)
```

**Parameters**

- **dateValue**
  - Type: `Date`

**Return Value**

Type: `Void`

**writeDateField(fieldName, dateValue)**

Writes a field name and value pair using the specified field name and date value. The date value is written in the ISO-8601 format.

**Signature**

```java
public Void writeDateField(String fieldName, Date dateValue)
```

**Parameters**

- **fieldName**
  - Type: `String`

- **dateValue**
  - Type: `Date`

**Return Value**

Type: `Void`
writeDateTime(datetimeValue)

Writes the specified date and time value in the ISO-8601 format.

Signature

public Void writeDateTime(Datetime datetimeValue)

Parameters

datetimeValue
  Type: Datetime

Return Value

Type: Void

writeDateTimeField(fieldName, datetimeValue)

Writes a field name and value pair using the specified field name and date and time value. The date and time value is written in the ISO-8601 format.

Signature

public Void writeDateTimeField(String fieldName, Datetime datetimeValue)

Parameters

fieldName
  Type: String
datetimeValue
  Type: Datetime

Return Value

Type: Void

writeEndArray()

Writes the ending marker of a JSON array (']').

Signature

public Void writeEndArray()

Return Value

Type: Void
writeEndObject()  
Writes the ending marker of a JSON object (').

Signature  
public Void writeEndObject()

Return Value  
Type: Void

writeFieldName(fieldName)  
Writes a field name.

Signature  
public Void writeFieldName(String fieldName)

Parameters  
fieldName  
Type: String

Return Value  
Type: Void

writeId(identifier)  
Writes the specified ID value.

Signature  
public Void writeId(ID identifier)

Parameters  
identifier  
Type: ID

Return Value  
Type: Void

writeIdField(fieldName, identifier)  
Writes a field name and value pair using the specified field name and identifier value.
Signature
public Void writeIdField(String fieldName, Id identifier)

Parameters
fieldName
  Type: String
identifier
  Type: ID

Return Value
Type: Void

writeNull()
Writes the JSON null literal value.

Signature
public Void writeNull()

Return Value
Type: Void

writeNullField(fieldName)
Writes a field name and value pair using the specified field name and the JSON null literal value.

Signature
public Void writeNullField(String fieldName)

Parameters
fieldName
  Type: String

Return Value
Type: Void

writeNumber(number)
Writes the specified decimal value.

Signature
public Void writeNumber(Decimal number)
Parameters

number
Type: Decimal

Return Value
Type: Void

writeNumber(number)
Writes the specified double value.

Signature
public Void writeNumber(Double number)

Parameters

number
Type: Double

Return Value
Type: Void

writeNumber(number)
Writes the specified integer value.

Signature
public Void writeNumber(Integer number)

Parameters

number
Type: Integer

Return Value
Type: Void

writeNumber(number)
Writes the specified long value.

Signature
public Void writeNumber(Long number)
Parameters

number
Type: Long

Return Value
Type: Void

`writeNumberField(fieldName, number)`
Writes a field name and value pair using the specified field name and decimal value.

**Signature**

```java
public Void writeNumberField(String fieldName, Decimal number)
```

**Parameters**

- `fieldName`
  Type: String

- `number`
  Type: Decimal

Return Value
Type: Void

`writeNumberField(fieldName, number)`
Writes a field name and value pair using the specified field name and double value.

**Signature**

```java
public Void writeNumberField(String fieldName, Double number)
```

**Parameters**

- `fieldName`
  Type: String

- `number`
  Type: Double

Return Value
Type: Void

`writeNumberField(fieldName, number)`
Writes a field name and value pair using the specified field name and integer value.
Signature
public Void writeNumberField(String fieldName, Integer number)

Parameters
fieldName
  Type: String
number
  Type: Integer

Return Value
Type: Void

writeNumberField(fieldName, number)
Writes a field name and value pair using the specified field name and long value.

Signature
public Void writeNumberField(String fieldName, Long number)

Parameters
fieldName
  Type: String
number
  Type: Long

Return Value
Type: Void

writeObject(anyObject)
Writes the specified Apex object in JSON format.

Signature
public Void writeObject(Object anyObject)

Parameters
anyObject
  Type: Object

Return Value
Type: Void
**writeObjectField(fieldName, value)**
Writes a field name and value pair using the specified field name and Apex object.

**Signature**
```
public Void writeObjectField(String fieldName, Object value)
```

**Parameters**
- `fieldName`
  - Type: String
- `value`
  - Type: Object

**Return Value**
Type: Void

**writeStartArray()**
Writes the starting marker of a JSON array ('[').

**Signature**
```
public Void writeStartArray()
```

**Return Value**
Type: Void

**writeStartObject()**
Writes the starting marker of a JSON object ('{').

**Signature**
```
public Void writeStartObject()
```

**Return Value**
Type: Void

**writeString(stringValue)**
Writes the specified string value.

**Signature**
```
public Void writeString(String stringValue)
```
Parameters

\[ \text{stringValue} \]
Type: String

Return Value
Type: Void

\textbf{writeStringField(fieldName, stringValue)}

Writes a field name and value pair using the specified field name and string value.

Signature

\textbf{public Void writeStringField(String fieldName, String stringValue)}

Parameters

\[ \text{fieldName} \]
Type: String

\[ \text{stringValue} \]
Type: String

Return Value
Type: Void

\textbf{writeTime(timeValue)}

Writes the specified time value in the ISO-8601 format.

Signature

\textbf{public Void writeTime(Time timeValue)}

Parameters

\[ \text{timeValue} \]
Type: Time

Return Value
Type: Void

\textbf{writeTimeField(fieldName, timeValue)}

Writes a field name and value pair using the specified field name and time value in the ISO-8601 format.
Signature

```java
public Void writeTimeField(String fieldName, Time timeValue)
```

Parameters

- **fieldName**
  - Type: `String`
- **timeValue**
  - Type: `Time`

Return Value

Type: `Void`

JSONParser Class

Represents a parser for JSON-encoded content.

Namespace

`System`

Usage

Use the `System.JSONParser` methods to parse a response that’s returned from a call to an external service that is in JSON format, such as a JSON-encoded response of a Web service callout.

SEE ALSO:

*Apex Developer Guide: JSON Parsing*

JSONParser Methods

The following are methods for `JSONParser`. All are instance methods.

IN THIS SECTION:

- `clearCurrentToken()`
  - Removes the current token.
- `getBlobValue()`
  - Returns the current token as a BLOB value.
- `getBooleanValue()`
  - Returns the current token as a Boolean value.
- `getCurrentName()`
  - Returns the name associated with the current token.
- `getCurrentToken()`
  - Returns the token that the parser currently points to or `null` if there's no current token.
getDatetimeValue()
Returns the current token as a date and time value.

getDateValue()
Returns the current token as a date value.

getDecimalValue()
Returns the current token as a decimal value.

getDoubleValue()
Returns the current token as a double value.

getIdValue()
Returns the current token as an ID value.

getIntegerValue()
Returns the current token as an integer value.

getLastClearedToken()
Returns the last token that was cleared by the clearCurrentToken method.

getLongValue()
Returns the current token as a long value.

getText()
Returns the textual representation of the current token or null if there’s no current token.

getTimeValue()
Returns the current token as a time value.

hasCurrentToken()
Returns true if the parser currently points to a token; otherwise, returns false.

nextToken()
Returns the next token or null if the parser has reached the end of the input stream.

nextValue()
Returns the next token that is a value type or null if the parser has reached the end of the input stream.

readValueAs(apexType)
Deserializes JSON content into an object of the specified Apex type and returns the deserialized object.

readValueAsStrict(apexType)
Deserializes JSON content into an object of the specified Apex type and returns the deserialized object. All attributes in the JSON content must be present in the specified type.

skipChildren()
Skips all child tokens of type JSONToken.START_ARRAY and JSONToken.START_OBJECT that the parser currently points to.

clearCurrentToken()
Removes the current token.

Signature

public Void clearCurrentToken()
## Return Value

**Type:** Void

**Usage**

After this method is called, a call to `hasCurrentToken` returns `false` and a call to `getCurrentToken` returns `null`. You can retrieve the cleared token by calling `getLastClearedToken`.

### `getBlobValue()`

Returns the current token as a BLOB value.

**Signature**

```java
public Blob getBlobValue()
```

**Return Value**

**Type:** Blob

**Usage**

The current token must be of type `JSONToken.VALUE_STRING` and must be Base64-encoded.

### `getBooleanValue()`

Returns the current token as a Boolean value.

**Signature**

```java
public Boolean getBooleanValue()
```

**Return Value**

**Type:** Boolean

**Usage**

The current token must be of type `JSONToken.VALUE_TRUE` or `JSONToken.VALUE_FALSE`.

The following example parses a sample JSON string and retrieves a Boolean value.

```java
String JSONContent = '{"isActive":true}';
JSONParser parser = JSON.createParser(JSONContent);
// Advance to the start object marker.
parser.nextToken();
// Advance to the next value.
parser.nextValue();
// Get the Boolean value.
Boolean isActive = parser.getBooleanValue();
```
getCurrentName()  
Returns the name associated with the current token.

Signature  
public String getCurrentName()  

Return Value  
Type: String  

Usage  
If the current token is of type JSONToken.FIELD_NAME, this method returns the same value as getText. If the current token is a value, this method returns the field name that precedes this token. For other values such as array values or root-level values, this method returns null.  
The following example parses a sample JSON string. It advances to the field value and retrieves its corresponding field name.

Example  
```java  
String JSONContent = '{"firstName":"John"}';  
JSONParser parser =  
    JSON.createParser(JSONContent);  
// Advance to the start object marker.  
parser.nextToken();  
// Advance to the next value.  
parser.nextValue();  
// Get the field name for the current value.  
String fieldName = parser.getCurrentName();  
// Get the textual representation  
// of the value.  
String fieldValue = parser.getText();  
```

getCurrentToken()  
Returns the token that the parser currently points to or null if there’s no current token.

Signature  
public System.JSONToken getCurrentToken()  

Return Value  
Type: System.JSONToken  

Usage  
The following example iterates through all the tokens in a sample JSON string.

Example  
```java  
String JSONContent = '{"firstName":"John"}';  
JSONParser parser =  
```
getDatetimeValue()

Returns the current token as a date and time value.

Signature

public Datetime getDatetimeValue()

Return Value

Type: Datetime

Usage

The current token must be of type JSONToken.VALUE_STRING and must represent a Datetime value in the ISO-8601 format.

The following example parses a sample JSON string and retrieves a Datetime value.

```java
String JSONContent = '{"transactionDate":"2011-03-22T13:01:23"}';
JSONParser parser = JSON.createParser(JSONContent);
// Advance to the start object marker.
parser.nextToken();
// Advance to the next value.
parsed.nextValue();
// Get the transaction date.
Datetime transactionDate = parser.getDatetimeValue();
```

generateValue()

Returns the current token as a date value.

Signature

public Date getDateValue()

Return Value

Type: Date

Usage

The current token must be of type JSONToken.VALUE_STRING and must represent a Date value in the ISO-8601 format.

```java
String JSONContent = '{"transactionDate":"2011-03-22T13:01:23"}';
JSONParser parser = JSON.createParser(JSONContent);
// Advance to the start object marker.
parsed.nextToken();
// Advance to the next value.
parsed.nextValue();
// Get the transaction date.
Datetime transactionDate = parsed.getDatetimeValue();
```
The following example parses a sample JSON string and retrieves a Date value.

```java
String JSONContent = 
   "{"dateOfBirth":"2011-03-22"};
JSONParser parser = 
   JSON.createParser(JSONContent);
// Advance to the start object marker.
parser.nextToken();
// Advance to the next value.
parser.nextValue();
// Get the date of birth.
Date dob = parser.getDateValue();
```

**getDecimalValue()**

Returns the current token as a decimal value.

**Signature**

```java
public Decimal getDecimalValue()
```

**Return Value**

Type: Decimal

**Usage**

The current token must be of type `JSONToken.VALUE_NUMBER_FLOAT` or `JSONToken.VALUE_NUMBER_INT` and is a numerical value that can be converted to a value of type `Decimal`.

The following example parses a sample JSON string and retrieves a Decimal value.

```java
String JSONContent = 
   "{"GPA":3.8}";
JSONParser parser = 
   JSON.createParser(JSONContent);
// Advance to the start object marker.
parser.nextToken();
// Advance to the next value.
parser.nextValue();
// Get the GPA score.
Decimal gpa = parser.getDecimalValue();
```

**getDoubleValue()**

Returns the current token as a double value.

**Signature**

```java
public Double getDoubleValue()
```
Return Value

Type: **Double**

Usage

The current token must be of type `JSONToken.VALUE_NUMBER_FLOAT` and is a numerical value that can be converted to a value of type **Double**.

The following example parses a sample JSON string and retrieves a Double value.

```java
String JSONContent = '
  "GPA":3.8"
';
JSONParser parser =
    JSON.createParser(JSONContent);
// Advance to the start object marker.
parser.nextToken();
// Advance to the next value.
parser.nextValue();
// Get the GPA score.
Double gpa = parser.getDoubleValue();
```

**getIdValue()**

Returns the current token as an ID value.

**Signature**

```java
public ID getIdValue()
```

**Return Value**

Type: **ID**

Usage

The current token must be of type `JSONToken.VALUE_STRING` and must be a valid **ID**.

The following example parses a sample JSON string and retrieves an ID value.

```java
String JSONContent = '
  "recordId":"001R0000002nO6H"
';
JSONParser parser =
    JSON.createParser(JSONContent);
// Advance to the start object marker.
parser.nextToken();
// Advance to the next value.
parser.nextValue();
// Get the record ID.
ID recordID = parser.getIdValue();
```

**getIntegerValue()**

Returns the current token as an integer value.
Signature

```java
public Integer getIntegerValue()
```

Return Value

Type: `Integer`

Usage

The current token must be of type `JSONToken.VALUE_NUMBER_INT` and must represent an `Integer`.

The following example parses a sample JSON string and retrieves an Integer value.

```java
String JSONContent = 
    '{"recordCount":10}';
JSONParser parser = 
    JSON.createParser(JSONContent);
// Advance to the start object marker.
parser.nextToken();
// Advance to the next value.
parser.nextValue();
// Get the record count.
Integer count = parser.getIntegerValue();
```

`getLastClearedToken()`

Returns the last token that was cleared by the `clearCurrentToken` method.

Signature

```java
public System.JSONToken getLastClearedToken()
```

Return Value

Type: `System.JSONToken`

`getLongValue()`

Returns the current token as a long value.

Signature

```java
public Long getLongValue()
```

Return Value

Type: `Long`

Usage

The current token must be of type `JSONToken.VALUE_NUMBER_INT` and is a numerical value that can be converted to a value of type `Long`. 
The following example parses a sample JSON string and retrieves a Long value.

```java
String JSONContent = 
    '{"recordCount":2097531021}';
JSONParser parser = 
    JSON.createParser(JSONContent);
// Advance to the start object marker.
parser.nextToken();
// Advance to the next value.
parser.nextValue();
// Get the record count.
Long count = parser.getLongValue();
```

**getText()**

Returns the textual representation of the current token or `null` if there's no current token.

**Signature**

```java
public String getText()
```

**Return Value**

Type: `String`

**Usage**

No current token exists, and therefore this method returns `null`, if `nextToken` has not been called yet for the first time or if the parser has reached the end of the input stream.

**getTimeValue()**

Returns the current token as a time value.

**Signature**

```java
public Time getTimeValue()
```

**Return Value**

Type: `Time`

**Usage**

The current token must be of type `JSONToken.VALUE_STRING` and must represent a `Time` value in the ISO-8601 format.

The following example parses a sample JSON string and retrieves a Datetime value.

```java
String JSONContent = 
    '{"arrivalTime":"18:05"}';
JSONParser parser = 
    JSON.createParser(JSONContent);
// Advance to the start object marker.
```
```java
token;
// Advance to the next value.
parser.nextValue();
// Get the arrival time.
Time arrivalTime = parser.getTimeValue();
```

**hasCurrentToken()**

Returns `true` if the parser currently points to a token; otherwise, returns `false`.

**Signature**

```java
public Boolean hasCurrentToken()
```

**Return Value**

Type: `Boolean`

**nextToken()**

Returns the next token or `null` if the parser has reached the end of the input stream.

**Signature**

```java
public System.JSONToken nextToken()
```

**Return Value**

Type: `System.JSONToken`

**Usage**

Advances the stream enough to determine the type of the next token, if any.

**nextValue()**

Returns the next token that is a value type or `null` if the parser has reached the end of the input stream.

**Signature**

```java
public System.JSONToken nextValue()
```

**Return Value**

Type: `System.JSONToken`

**Usage**

Advances the stream enough to determine the type of the next token that is of a value type, if any, including a JSON array and object start and end markers.
**readValueAs (apexType)**

Deserializes JSON content into an object of the specified Apex type and returns the deserialized object.

**Signature**

```java
public Object readValueAs(System.Type apexType)
```

**Parameters**

- **apexType**
  - Type: `System.Type`
  - The `apexType` argument specifies the type of the object that this method returns after deserializing the current value.

**Return Value**

Type: `Object`

**Usage**

If the JSON content contains attributes not present in the `System.Type` argument, such as a missing field or object, deserialization fails in some circumstances. When deserializing JSON content into a custom object or an sObject using Salesforce API version 34.0 or earlier, this method throws a runtime exception when passed extraneous attributes. When deserializing JSON content into an Apex class in any API version, or into an object in API version 35.0 or later, no exception is thrown. When no exception is thrown, this method ignores extraneous attributes and parses the rest of the JSON content.

**Example**

The following example parses a sample JSON string and retrieves a Datetime value. Before being able to run this sample, you must create a new Apex class as follows:

```java
public class Person {
    public String name;
    public String phone;
}
```

Next, insert the following sample in a class method:

```java
// JSON string that contains a Person object.
String JSONContent =
    '{"person":{
        "name":"John Smith",
        "phone":"555-1212")};
JSONParser parser =
    JSON.createParser(JSONContent);
// Make calls to nextToken()
// to point to the second
// start object marker.
parser.nextToken();
parser.nextToken();
parser.nextToken();
// Retrieve the Person object
// from the JSON string.
```
Person obj = (Person)parser.readValueAs(Person.class);
System.assertEquals(obj.name, 'John Smith');
System.assertEquals(obj.phone, '555-1212');

**readValueAsStrict(apexType)**

Deserializes JSON content into an object of the specified Apex type and returns the deserialized object. All attributes in the JSON content must be present in the specified type.

**Signature**

```java
public Object readValueAsStrict(System.Type apexType)
```

**Parameters**

`apexType`

Type: `System.Type`

The `apexType` argument specifies the type of the object that this method returns after deserializing the current value.

**Return Value**

Type: Object

**Usage**

If the JSON content contains attributes not present in the `System.Type` argument, such as a missing field or object, deserialization fails in some circumstances. When deserializing JSON content with extraneous attributes into an Apex class, this method throws an exception in all API versions. However, no exception is thrown when you use this method to deserialize JSON content into a custom object or an sObject.

The following example parses a sample JSON string and retrieves a Datetime value. Before being able to run this sample, you must create a new Apex class as follows:

```java
public class Person {
    public String name;
    public String phone;
}
```

Next, insert the following sample in a class method:

```java
// JSON string that contains a Person object.
String JSONContent = '{"person":{"name":"John Smith","phone":"555-1212"}}';
JSONParser parser = JSON.createParser(JSONContent);
// Make calls to nextToken()
// to point to the second
```
// start object marker.
parser.nextToken();
parser.nextToken();
parser.nextToken();

// Retrieve the Person object
// from the JSON string.
Person obj =
    (Person)parser.readValueAsStrict(
            Person.class);
System.assertEquals(
    obj.name, 'John Smith');
System.assertEquals(
    obj.phone, '555-1212');

skipChildren()

Skips all child tokens of type JSONToken.START_ARRAY and JSONToken.START_OBJECT that the parser currently points to.

Signature

public Void skipChildren()

Return Value

Type: Void

JSONToken Enum

Contains all token values used for parsing JSON content.

Namespace

System

<table>
<thead>
<tr>
<th>Enum Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>END_ARRAY</td>
<td>The ending of an array value. This token is returned when ']' is encountered.</td>
</tr>
<tr>
<td>END_OBJECT</td>
<td>The ending of an object value. This token is returned when '}' is encountered.</td>
</tr>
<tr>
<td>FIELD_NAME</td>
<td>A string token that is a field name.</td>
</tr>
<tr>
<td>NOT_AVAILABLE</td>
<td>The requested token isn't available.</td>
</tr>
<tr>
<td>START_ARRAY</td>
<td>The start of an array value. This token is returned when '[' is encountered.</td>
</tr>
<tr>
<td>START_OBJECT</td>
<td>The start of an object value. This token is returned when '{' is encountered.</td>
</tr>
</tbody>
</table>

3160
<table>
<thead>
<tr>
<th>Enum Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>VALUE_EMBEDDED_OBJECT</td>
<td>An embedded object that isn’t accessible as a typical object structure that includes the start and end object tokens START_OBJECT and END_OBJECT but is represented as a raw object.</td>
</tr>
<tr>
<td>VALUE_FALSE</td>
<td>The literal &quot;false&quot; value.</td>
</tr>
<tr>
<td>VALUE_NULL</td>
<td>The literal &quot;null&quot; value.</td>
</tr>
<tr>
<td>VALUE_NUMBER_FLOAT</td>
<td>A float value.</td>
</tr>
<tr>
<td>VALUE_NUMBER_INT</td>
<td>An integer value.</td>
</tr>
<tr>
<td>VALUE_STRING</td>
<td>A string value.</td>
</tr>
<tr>
<td>VALUE_TRUE</td>
<td>A value that corresponds to the &quot;true&quot; string literal.</td>
</tr>
</tbody>
</table>

**Label Class**

Provides methods to retrieve a custom label or to check if translation exists for a label in a specific language and namespace. Label names are dynamically resolved at run time, overriding the user’s current language if a translation exists for the requested language. You can’t access labels that are protected in a different namespace.

**Namespace**

*System*

**Usage**

Custom labels enable developers to create multilingual applications by automatically presenting information (for example, help text or error messages) in a user’s native language. Custom labels have a limit of 1000 characters and can be accessed from Apex classes, Visualforce pages, Lightning pages, or Lightning components. For more information on custom labels, see Custom Labels in Salesforce Help. The label values can be translated into any language Salesforce supports. For supported languages, see Supported Languages in Salesforce Help.

- To define custom labels, from Setup, in the Quick Find box, enter *Custom Labels*, and then select *Custom Labels*.
- To assign translated values, turn on Translation Workbench and add translation mappings.
- To retrieve the label for a default language setting or for a language and namespace, use `System.Label.get(namespace, label, language)`.
- To check if translation exists for a label and language in a namespace, use `Label.translationExists(namespace, label, language)`.

In Apex code, you can refer to or instantiate a Label like this:

```
System.Label.myLabelName
```

For information on passing in labels into Aura components, see Getting Labels in Apex in the Lightning Aura Components Developer Guide.
Examples

This example returns True if an English label called "MyLabel" exists in the "MyNamespace" namespace.

```java
boolean exists = Label.translationExists('MyNamespace', 'MyLabel', 'en')
```

This example retrieves the custom label translation text for "MyLabel" in French.

```java
String value = Label.get('MyNamespace', 'MyLabel', 'fr')
```

IN THIS SECTION:

Label Methods

The following are methods for Label.

IN THIS SECTION:

get(namespace, label)
Retrieve a custom label for the specified namespace and a default language setting.

get(namespace, label, language)
Retrieve a custom label for the specified namespace and language.

translationExists(namespace, label, language)
Check if translation exists for a label and language in a namespace.

**get(namespace, label)**

Retrieve a custom label for the specified namespace and a default language setting.

**Signature**

```java
public static String get(String namespace, String label)
```

**Parameters**

- `namespace`
  - Type: `String`
    - If the namespace name is null, it defaults to the package namespace. If the namespace name is an empty string, it implies the org namespace.

- `label`
  - Type: `String`
    - The label name cannot be null or an empty string.

**Return Value**

- Type: `String`
**get(namespace, label, language)**
Retrieve a custom label for the specified namespace and language.

**Signature**
```java
public static String get(String namespace, String label, String language)
```

**Parameters**
- **namespace**
  - Type: `String`
  - If the namespace name is null, it defaults to the package namespace. If the namespace name is an empty string, it implies the org namespace.
- **label**
  - Type: `String`
  - The label name cannot be null or an empty string.
- **language**
  - Type: `String`
  - This parameter must be a valid language ISO code. See the Platform-Only Languages section in Supported Languages in Salesforce Help.

**Return Value**
- Type: `String`

**translationExists(namespace, label, language)**
Check if translation exists for a label and language in a namespace.

**Signature**
```java
public static Boolean translationExists(string namespace, string label, string language)
```

**Parameters**
- **namespace**
  - Type: `String`
  - If the namespace name is null, it defaults to the package namespace. If the namespace name is an empty string, it implies the org namespace.
- **label**
  - Type: `String`
  - The label name cannot be null or an empty string.
- **language**
  - Type: `String`
  - This parameter must be a valid language ISO code. See the Platform-Only Languages section in Supported Languages in Salesforce Help.
Return Value
Type: Boolean

Limits Class
Contains methods that return limit information for specific resources.

Namespace
System

Usage
The Limits methods return the specific limit for the particular governor, such as the number of calls of a method or the amount of heap size remaining.

Because Apex runs in a multitenant environment, the Apex runtime engine strictly enforces a number of limits to ensure that runaway Apex doesn’t monopolize shared resources.

None of the Limits methods require an argument. The format of the limits methods is as follows:

```
myDMLLimit = Limits.getDMLStatements();
```

There are two versions of every method: the first returns the amount of the resource that has been used while the second version contains the word limit and returns the total amount of the resource that is available.

See Execution Governors and Limits.

Limits Methods
The following are methods for Limits. All methods are static.

IN THIS SECTION:
- `getAggregateQueries()`: Returns the number of aggregate queries that have been processed with any SOQL query statement.
- `getLimitAggregateQueries()`: Returns the total number of aggregate queries that can be processed with SOQL query statements.
- `getAsynCalls()`: Reserved for future use.
- `getLimitAsynCalls()`: Reserved for future use.
- `getCallouts()`: Returns the number of Web service statements that have been processed.
- `getChildRelationshipsDescribes()`: Deprecated. Returns the number of child relationship objects that have been returned.
- `getLimitCallouts()`: Returns the total number of Web service statements that can be processed.
getCpuTime()  
Returns the CPU time (in milliseconds) that has been used in the current transaction.

gLimitCpuTime()  
Returns the maximum CPU time (in milliseconds) that can be used in a transaction.

gDMLRows()  
Returns the number of records that have been processed with any statement that counts against DML limits, such as DML statements, the Database.emptyRecycleBin method, and other methods.

gLimitDMLRows()  
Returns the total number of records that can be processed with any statement that counts against DML limits, such as DML statements, the database.EmptyRecycleBin method, and other methods.

gDMLStatements()  
Returns the number of DML statements (such as insert, update or the database.EmptyRecycleBin method) that have been called.

gLimitDMLStatements()  
Returns the total number of DML statements or the database.EmptyRecycleBin methods that can be called.

gEmailInvocations()  
Returns the number of email invocations (such as sendEmail) that have been called.

gLimitEmailInvocations()  
Returns the total number of email invocation (such as sendEmail) that can be called.

gFindSimilarCalls()  
Deprecated. Returns the same value as getSoslQueries. The number of findSimilar methods is no longer a separate limit, but is tracked as the number of SOSL queries issued.

gLimitFindSimilarCalls()  
Deprecated. Returns the same value as getLimitSoslQueries. The number of findSimilar methods is no longer a separate limit, but is tracked as the number of SOSL queries issued.

gFutureCalls()  
Returns the number of methods with the future annotation that have been executed (not necessarily completed).

gLimitFutureCalls()  
Returns the total number of methods with the future annotation that can be executed (not necessarily completed).

gHeapSize()  
Returns the approximate amount of memory (in bytes) that has been used for the heap.

gLimitHeapSize()  
Returns the total amount of memory (in bytes) that can be used for the heap.

gMobilePushApexCalls()  
Returns the number of Apex calls that have been used by mobile push notifications during the current metering interval.

gLimitMobilePushApexCalls()  
Returns the total number of Apex calls that are allowed per transaction for mobile push notifications.

gPublishImmediateDML()  
Returns the number of EventBus.publish calls that have been made for platform events configured to publish immediately.
getLimitPublishImmediateDML()
Returns the total number of EventBus.publish statements that can be called for platform events configured to publish immediately.

getQueries()
Returns the number of SOQL queries that have been issued.

gGetLimitQueries()
Returns the total number of SOQL queries that can be issued.

ggetQueryLocatorRows()
Returns the number of records that have been returned by the Database.getQueryLocator method.

gGetLimitQueryLocatorRows()
Returns the total number of records that can be returned by the Database.getQueryLocator method.

ggetQueryRows()
Returns the number of records that have been returned by issuing SOQL queries.

gGetLimitQueryRows()
Returns the total number of records that can be returned by issuing SOQL queries.

gQueueableJobs()
Returns the number of queueable jobs that have been added to the queue per transaction. A queueable job corresponds to a class that implements the Queueable interface.

gGetLimitQueueableJobs()
Returns the maximum number of queueable jobs that can be added to the queue per transaction. A queueable job corresponds to a class that implements the Queueable interface.

gerunAs()
Deprecated. Returns the same value as getDMLStatements.

gerunAs()
Deprecated. Returns the same value as getLimitDMLStatements.

gSavepointRollbacks()
Deprecated. Returns the same value as getDMLStatements.

gGetSavepointRollbacks()
Deprecated. Returns the same value as getLimitDMLStatements.

gSavepoints()
Deprecated. Returns the same value as getDMLStatements.

gGetSavepoints()
Deprecated. Returns the same value as getLimitDMLStatements.

gSoslQueries()
Returns the number of SOSL queries that have been issued.

gGetLimitSoslQueries()
Returns the total number of SOSL queries that can be issued.

gAgggregateQueries()
Returns the number of aggregate queries that have been processed with any SOQL query statement.
Signature

public static Integer getAggregateQueries()

Return Value
Type: Integer

getLimitAggregateQueries()
Returns the total number of aggregate queries that can be processed with SOQL query statements.

Signature

public static Integer getLimitAggregateQueries()

Return Value
Type: Integer

getAsynccalls()
Reserved for future use.

Signature

public static Integer getAsynccalls()

Return Value
Type: Integer

getLimitAsynccalls()
Reserved for future use.

Signature

public static Integer getLimitAsynccalls()

Return Value
Type: Integer

getCallouts()
Returns the number of Web service statements that have been processed.

Signature

public static Integer getCallouts()
Return Value
Type: Integer

getchildrelationshipsdescribes()
Deprecated. Returns the number of child relationship objects that have been returned.

Signature
public static Integer getchildrelationshipsdescribes()

Return Value
Type: Integer

Usage
Note: Because describe limits are no longer enforced in any API version, this method is no longer available. In API version 30.0 and earlier, this method is available but is deprecated.

gelimitcallouts()
Returns the total number of Web service statements that can be processed.

Signature
public static Integer gelimitcallouts()

Return Value
Type: Integer

gecputime()
Returns the CPU time (in milliseconds) that has been used in the current transaction.

Signature
public static Integer gecputime()

Return Value
Type: Integer

gelimitcputime()
Returns the maximum CPU time (in milliseconds) that can be used in a transaction.
Signature

```java
public static Integer getLimitCpuTime()
```

Return Value

Type: Integer

**getDMLRows()**

Returns the number of records that have been processed with any statement that counts against DML limits, such as DML statements, the `Database.emptyRecycleBin` method, and other methods.

Signature

```java
public static Integer getDMLRows()
```

Return Value

Type: Integer

**getLimitDMLRows()**

Returns the total number of records that can be processed with any statement that counts against DML limits, such as DML statements, the `database.EmptyRecycleBin` method, and other methods.

Signature

```java
public static Integer getLimitDMLRows()
```

Return Value

Type: Integer

**getDMLStatements()**

Returns the number of DML statements (such as `insert`, `update` or the `database.EmptyRecycleBin` method) that have been called.

Signature

```java
public static Integer getDMLStatements()
```

Return Value

Type: Integer

**getLimitDMLStatements()**

Returns the total number of DML statements or the `Database.EmptyRecycleBin` methods that can be called.
Signature
public static Integer getLimitDMLStatements()

Return Value
Type: Integer

getEmailInvocations()
Returns the number of email invocations (such as sendEmail) that have been called.

Signature
public static Integer getEmailInvocations()

Return Value
Type: Integer

getLimitEmailInvocations()
Returns the total number of email invocation (such as sendEmail) that can be called.

Signature
public static Integer getLimitEmailInvocations()

Return Value
Type: Integer

getFindSimilarCalls()
Deprecated. Returns the same value as getSoslQueries. The number of findSimilar methods is no longer a separate limit, but is tracked as the number of SOSL queries issued.

Signature
public static Integer getFindSimilarCalls()

Return Value
Type: Integer

getLimitFindSimilarCalls()
Deprecated. Returns the same value as getLimitSoslQueries. The number of findSimilar methods is no longer a separate limit, but is tracked as the number of SOSL queries issued.
Signature

```
public static Integer getLimitFindSimilarCalls()
```

Return Value
Type: Integer

**getFutureCalls()**
Returns the number of methods with the `future` annotation that have been executed (not necessarily completed).

Signature

```
public static Integer getFutureCalls()
```

Return Value
Type: Integer

**getLimitFutureCalls()**
Returns the total number of methods with the `future` annotation that can be executed (not necessarily completed).

Signature

```
public static Integer getLimitFutureCalls()
```

Return Value
Type: Integer

**getHeapSize()**
Returns the approximate amount of memory (in bytes) that has been used for the heap.

Signature

```
public static Integer getHeapSize()
```

Return Value
Type: Integer

**getLimitHeapSize()**
Returns the total amount of memory (in bytes) that can be used for the heap.

Signature

```
public static Integer getLimitHeapSize()
```
Return Value
Type: Integer

getMobilePushApexCalls()
Returns the number of Apex calls that have been used by mobile push notifications during the current metering interval.

Signature
public static Integer getMobilePushApexCalls()

Return Value
Type: Integer

gLimitMobilePushApexCalls()
Returns the total number of Apex calls that are allowed per transaction for mobile push notifications.

Signature
public static Integer getLimitMobilePushApexCalls()

Return Value
Type: Integer

getPublishImmediateDML()
Returns the number of EventBus.publish calls that have been made for platform events configured to publish immediately.

Signature
public static Integer getPublishImmediateDML()

Return Value
Type: Integer

getLimitPublishImmediateDML()
Returns the total number of EventBus.publish statements that can be called for platform events configured to publish immediately.

Signature
public static Integer getLimitPublishImmediateDML()

Return Value
Type: Integer
getQueries()
Returns the number of SOQL queries that have been issued.

Signature
public static Integer getQueries()

Return Value
Type: Integer

genLimitQueries()
Returns the total number of SOQL queries that can be issued.

Signature
public static Integer getLimitQueries()

Return Value
Type: Integer

getQueryLocatorRows()
Returns the number of records that have been returned by the Database.getQueryLocator method.

Signature
public static Integer getQueryLocatorRows()

Return Value
Type: Integer

getLimitQueryLocatorRows()
Returns the total number of records that can be returned by the Database.getQueryLocator method.

Signature
public static Integer getLimitQueryLocatorRows()

Return Value
Type: Integer

getQueryRows()
Returns the number of records that have been returned by issuing SOQL queries.
### getQueryRows()

Returns the total number of records that can be returned by issuing SOQL queries.

**Signature**

```java
generate public static Integer getQueryRows()
```

**Return Value**

Type: `Integer`

### getLimitQueryRows()

Returns the maximum number of records that can be returned by issuing SOQL queries.

**Signature**

```java
generate public static Integer getLimitQueryRows()
```

**Return Value**

Type: `Integer`

### getQueueableJobs()

Returns the number of queueable jobs that have been added to the queue per transaction. A queueable job corresponds to a class that implements the `Queueable` interface.

**Signature**

```java
generate public static Integer getQueueableJobs()
```

**Return Value**

Type: `Integer`

### getLimitQueueableJobs()

Returns the maximum number of queueable jobs that can be added to the queue per transaction. A queueable job corresponds to a class that implements the `Queueable` interface.

**Signature**

```java
generate public static Integer getLimitQueueableJobs()
```

**Return Value**

Type: `Integer`

### getRunAs()

Deprecated. Returns the same value as `getDMLStatements`.

**Signature**

```java
generate public static Integer getRunAs()
```

**Return Value**

Type: `Integer`
Signature

```java
public static Integer getRunAs()
```

Return Value

Type: Integer

Usage

The number of `RunAs` methods is no longer a separate limit, but is tracked as the number of DML statements issued.

```java
getLimitRunAs()
```

Deprecated. Returns the same value as `getLimitDMLStatements`.

Signature

```java
public static Integer getLimitRunAs()
```

Return Value

Type: Integer

Usage

The number of `RunAs` methods is no longer a separate limit, but is tracked as the number of DML statements issued.

```java
getSavepointRollbacks()
```

Deprecated. Returns the same value as `getDMLStatements`.

Signature

```java
public static Integer getSavepointRollbacks()
```

Return Value

Type: Integer

Usage

The number of `Rollback` methods is no longer a separate limit, but is tracked as the number of DML statements issued.

```java
getLimitSavepointRollbacks()
```

Deprecated. Returns the same value as `getLimitDMLStatements`.

Signature

```java
public static Integer getLimitSavepointRollbacks()
```
Return Value
Type: Integer

Usage
The number of Rollback methods is no longer a separate limit, but is tracked as the number of DML statements issued.

getsavepoints()
Deprecated. Returns the same value as getDMLStatements.

Signature
public static Integer getsavepoints()

Return Value
Type: Integer

Usage
The number of setSavepoint methods is no longer a separate limit, but is tracked as the number of DML statements issued.

getLimitsavepoints()
Deprecated. Returns the same value as getLimitDMLStatements.

Signature
public static Integer getLimitsavepoints()

Return Value
Type: Integer

Usage
The number of setSavepoint methods is no longer a separate limit, but is tracked as the number of DML statements issued.

getSoslQueries()
Returns the number of SOSL queries that have been issued.

Signature
public static Integer getSoslQueries()

Return Value
Type: Integer
**getLimitSoslQueries()**

Returns the total number of SOSL queries that can be issued.

**Signature**

```java
public static Integer getLimitSoslQueries()
```

**Return Value**

*Type:* Integer

**List Class**

Contains methods for the List collection type.

**Namespace**

*System*

**Usage**

The list methods are all instance methods, that is, they operate on a particular instance of a list. For example, the following removes all elements from `myList`:

```java
myList.clear();
```

Even though the `clear` method does not include any parameters, the list that calls it is its implicit parameter.

![Note:](https://example.com)

- When using a custom type for the list elements, provide an `equals` method in your class. Apex uses this method to determine equality and uniqueness for your objects. For more information on providing an `equals` method, see [Using Custom Types in Map Keys and Sets](https://www.salesforce.com/

- If the list contains String elements, the elements are case-sensitive. Two list elements that differ only by case are considered distinct.

For more information on lists, see [Lists](https://www.salesforce.com/).

**IN THIS SECTION:**

- List Constructors
- List Methods

**List Constructors**

The following are constructors for List.
IN THIS SECTION:

List<T>()
Creates a new instance of the List class. A list can hold elements of any data type T.

List<T>(listToCopy)
Creates a new instance of the List class by copying the elements from the specified list. T is the data type of the elements in both lists and can be any data type.

List<T>(setToCopy)
Creates a new instance of the List class by copying the elements from the specified set. T is the data type of the elements in the set and list and can be any data type.

List<T>()
Creates a new instance of the List class. A list can hold elements of any data type T.

Signature
public List<T>()

Example

```java
// Create a list
List<Integer> ls1 = new List<Integer>();
// Add two integers to the list
ls1.add(1);
ls1.add(2);
```

List<T>(listToCopy)
Creates a new instance of the List class by copying the elements from the specified list. T is the data type of the elements in both lists and can be any data type.

Signature
public List<T>(List<T> listToCopy)

Parameters

listToCopy
Type: List<T>
The list containing the elements to initialize this list from. T is the data type of the list elements.

Example

```java
List<Integer> ls1 = new List<Integer>();
ls1.add(1);
ls1.add(2);
// Create a list based on an existing one
List<Integer> ls2 = new List<Integer>(ls1);
```
List\langle T\rangle (setToCopy)

Creates a new instance of the List class by copying the elements from the specified set. T is the data type of the elements in the set and list and can be any data type.

Signature

\textbf{public} List\langle T\rangle (Set\langle T\rangle setToCopy)

Parameters

setToCopy

Type: Set\langle T\rangle

The set containing the elements to initialize this list with. T is the data type of the set elements.

Example

```java
Set<Integer> s1 = new Set<Integer>();
set.add(1);
set.add(2);
// Create a list based on a set
List<Integer> ls = new List<Integer>(s1);
// ls elements are copied from s1
System.debug(ls);// DEBUG\langle 1, 2\rangle
```

List Methods

The following are methods for List. All are instance methods.

\textbf{IN THIS SECTION:}

\begin{itemize}
\item \textbf{add(listElement)}
\end{itemize}

Adds an element to the end of the list.

\begin{itemize}
\item \textbf{add(index, listElement)}
\end{itemize}

Inserts an element into the list at the specified index position.

\begin{itemize}
\item \textbf{addAll(fromList)}
\end{itemize}

Adds all of the elements in the specified list to the list that calls the method. Both lists must be of the same type.

\begin{itemize}
\item \textbf{addAll(fromSet)}
\end{itemize}

Add all of the elements in specified set to the list that calls the method. The set and the list must be of the same type.

\begin{itemize}
\item \textbf{clear()}
\end{itemize}

Removes all elements from a list, consequently setting the list’s length to zero.

\begin{itemize}
\item \textbf{clone()}
\end{itemize}

Makes a duplicate copy of a list.
contains(listElement)
Returns true if the list contains the specified element.

deepClone(preserveId, preserveReadonlyTimestamps, preserveAutonumber)
Makes a duplicate copy of a list of sObject records, including the sObject records themselves.
equals(list2)
Compares this list with the specified list and returns true if both lists are equal; otherwise, returns false.

get(index)
Returns the list element stored at the specified index.

getSObjectType()
Returns the token of the sObject type that makes up a list of sObjects.

hashCode()
Returns the hashcode corresponding to this list and its contents.

indexOf(listElement)
Returns the index of the first occurrence of the specified element in this list. If this list does not contain the element, returns -1.

isEmpty()
Returns true if the list has zero elements.

iterator()
Returns an instance of an iterator for this list.

remove(index)
Removes the list element stored at the specified index, returning the element that was removed.

set(index, listElement)
Sets the specified value for the element at the given index.

size()
Returns the number of elements in the list.

sort()
Sorts the items in the list in ascending order.

toString()
Returns the string representation of the list.

**add(listElement)**
Adds an element to the end of the list.

**Signature**

```java
public Void add(Object listElement)
```

**Parameters**

`listElement`
Type: Object
Return Value
Type: Void

Example
```
List<Integer> myList = new List<Integer>(){
    myList.add(47);
    Integer myNumber = myList.get(0);
    system.assertEquals(47, myNumber);
}
```

add(index, listElement)
Inserts an element into the list at the specified index position.

Signature
```
public Void add(Integer index, Object listElement)
```

Parameters
- index
  Type: Integer
- listElement
  Type: Object

Return Value
Type: Void

Example
In the following example, a list with six elements is created, and integers are added to the first and second index positions.
```
List<Integer> myList = new Integer[6];
myList.add(0, 47);
myList.add(1, 52);
system.assertEquals(52, myList.get(1));
```

addAll(fromList)
Adds all of the elements in the specified list to the list that calls the method. Both lists must be of the same type.

Signature
```
public Void addAll(List fromList)
```

Parameters
- fromList
  Type: List
Return Value
Type: Void

**addAll(fromSet)**
Add all of the elements in specified set to the list that calls the method. The set and the list must be of the same type.

**Signature**

```java
public Void addAll(Set fromSet)
```

**Parameters**

*fromSet*
Type: Set

**Return Value**
Type: Void

**clear()**
Removes all elements from a list, consequently setting the list’s length to zero.

**Signature**

```java
public Void clear()
```

**Return Value**
Type: Void

**clone()**
Makes a duplicate copy of a list.

**Signature**

```java
public List<Object> clone()
```

**Return Value**
Type: List<Object>

**Usage**
The cloned list is of the same type as the current list.
Note that if this is a list of sObject records, the duplicate list will only be a shallow copy of the list. That is, the duplicate will have references to each object, but the sObject records themselves will not be duplicated. For example:
To also copy the sObject records, you must use the `deepClone` method.
Example

```java
Account a = new Account(Name='Acme', BillingCity='New York');
Account b = new Account();
Account[] q1 = new Account[]{a, b};
Account[] q2 = q1.clone();
q1[0].BillingCity = 'San Francisco';

System.assertEquals('San Francisco', q1[0].BillingCity);
System.assertEquals('San Francisco', q2[0].BillingCity);
```

`contains(listElement)`

Returns `true` if the list contains the specified element.

Signature

```java
public Boolean contains(Object listElement)
```

Parameters

`listElement`
Type: `Object`

Return Value

Type: `Boolean`

Example

```java
List<String> myStrings = new List<String>{'a', 'b'};
Boolean result = myStrings.contains('z');
System.assertEquals(false, result);
```

deepClone(preserveId, preserveReadonlyTimestamps, preserveAutonumber)

Makes a duplicate copy of a list of sObject records, including the sObject records themselves.

Signature

```java
public List<Object> deepClone(Boolean preserveId, Boolean preserveReadonlyTimestamps, Boolean preserveAutonumber)
```
Parameters

*preserveId*
Type: Boolean
The optional *preserveId* argument determines whether the IDs of the original objects are preserved or cleared in the duplicates. If set to *true*, the IDs are copied to the cloned objects. The default is *false*, that is, the IDs are cleared.

*preserveReadonlyTimestamps*
Type: Boolean
The optional *preserveReadonlyTimestamps* argument determines whether the read-only timestamp and user ID fields are preserved or cleared in the duplicates. If set to *true*, the read-only fields CreatedById, CreatedDate, LastModifiedById, and LastModifiedDate are copied to the cloned objects. The default is *false*, that is, the values are cleared.

*preserveAutonumber*
Type: Boolean
The optional *preserveAutonumber* argument determines whether the autonumber fields of the original objects are preserved or cleared in the duplicates. If set to *true*, auto number fields are copied to the cloned objects. The default is *false*, that is, auto number fields are cleared.

Return Value
Type: List<Object>

Usage
The returned list is of the same type as the current list.

Note:
- `deepClone` only works with lists of sObjects, not with lists of primitives.
- For Apex saved using Salesforce API version 22.0 or earlier, the default value for the *preserve_id* argument is *true*, that is, the IDs are preserved.

To make a shallow copy of a list without duplicating the sObject records it contains, use the `clone` method.

Example
This example performs a deep clone for a list with two accounts.

```java
Account a = new Account(Name='Acme', BillingCity='New York');
Account b = new Account(Name='Salesforce');
Account[] q1 = new Account[]{a,b};
Account[] q2 = q1.deepClone();
q1[0].BillingCity = 'San Francisco';
System.assertEquals('San Francisco', q1[0].BillingCity);
```
This example is based on the previous example and shows how to clone a list with preserved read-only timestamp and user ID fields.

```
insert q1;

List<Account> accts = [SELECT CreatedById, CreatedDate, LastModifiedById, LastModifiedDate, BillingCity
FROM Account
WHERE Name='Acme' OR Name='Salesforce'];

// Clone list while preserving timestamp and user ID fields.
Account[] q3 = accts.deepClone(false, true, false);

// Verify timestamp fields are preserved for the first list element.
System.assertEquals(accts[0].CreatedById, q3[0].CreatedById);
System.assertEquals(accts[0].CreatedDate, q3[0].CreatedDate);
System.assertEquals(accts[0].LastModifiedById, q3[0].LastModifiedById);
System.assertEquals(accts[0].LastModifiedDate, q3[0].LastModifiedDate);
```

### equals(list2)

Compares this list with the specified list and returns `true` if both lists are equal; otherwise, returns `false`.

**Signature**

```
public Boolean equals(List list2)
```

**Parameters**

- **list2**
  - Type: `List`
  - The list to compare this list with.

**Return Value**

- Type: `Boolean`

**Usage**

Two lists are equal if their elements are equal and are in the same order. The `==` operator is used to compare the elements of the lists.
The `==` operator is equivalent to calling the `equals` method, so you can call `list1.equals(list2)` instead of `list1 == list2`.

**get(index)**

Returns the list element stored at the specified index.

**Signature**

```java
public Object get(Integer index)
```

**Parameters**

- `index`
  
  Type: `Integer`

**Return Value**

Type: `Object`

**Usage**

To reference an element of a one-dimensional list of primitives or sObjects, you can also follow the name of the list with the element’s index position in square brackets as shown in the example.

**Example**

```java
List<Integer> myList = new List<Integer>();
myList.add(47);
Integer myNumber = myList.get(0);
system.assertEquals(47, myNumber);
```

```java
List<String> colors = new String[3];
colors[0] = 'Red';
colors[1] = 'Blue';
colors[2] = 'Green';
```

**getSObjectType()**

Returns the token of the sObject type that makes up a list of sObjects.

**Signature**

```java
public Schema.SObjectType getSObjectType()
```

**Return Value**

Type: `Schema.SObjectType`
Usage

Use this method with describe information to determine if a list contains sObjects of a particular type. Note that this method can only be used with lists that are composed of sObjects. For more information, see Understanding Apex Describe Information.

Example

```java
// Create a generic sObject variable.
SObject sObj = Database.query('SELECT Id FROM Account LIMIT 1');

// Verify if that sObject variable is an Account token.
System.assertEquals(Account.sObjectType, sObj.getSObjectType());

// Create a list of generic sObjects.
List<sObject> q = new Account[]{};

// Verify if the list of sObjects contains Account tokens.
System.assertEquals(Account.sObjectType, q.getSObjectType());
```

hashCode()

Returns the hashcode corresponding to this list and its contents.

Signature

```java
public Integer hashCode()
```

Return Value

Type: Integer

indexOf(listElement)

Returns the index of the first occurrence of the specified element in this list. If this list does not contain the element, returns -1.

Signature

```java
public Integer indexOf(Object listElement)
```

Parameters

```java
listElement
    Type: Object
```
Return Value
Type: Integer

Example
```
List<String> myStrings = new List<String>{'a', 'b', 'a'};
Integer result = myStrings.indexOf('a');
System.assertEquals(0, result);
```

isEmpty()
Returns true if the list has zero elements.

Signature
```
public Boolean isEmpty()
```

Return Value
Type: Boolean

iterator()
Returns an instance of an iterator for this list.

Signature
```
public Iterator iterator()
```

Usage
From the returned iterator, you can use the iterable methods hasNext and next to iterate through the list.

Note: You don’t have to implement the iterable interface to use the iterable methods with a list.

See Custom Iterators.

Example
```
public class CustomIterator
    implements Iterator<Account>{

    private List<Account> accounts;
    private Integer currentIndex;

    public CustomIterator(List<Account> accounts){
        this.accounts = accounts;
```
this.currentIndex = 0;

public Boolean hasNext(){
    return currentIndex < accounts.size();
}

public Account next(){
    if(hasNext()) {
        return accounts[currentIndex++];
    } else {
        throw new NoSuchElementException('Iterator has no more elements.');
    }
}

remove(index)
Removes the list element stored at the specified index, returning the element that was removed.

Signature
public Object remove(Integer index)

Parameters
index
Type: Integer

Return Value
Type: Object

Example
List<String> colors = new String[3];
colors[0] = 'Red';
colors[1] = 'Blue';
colors[2] = 'Green';
String s1 = colors.remove(2);
System.assertEquals('Green', s1);

set(index, listElement)
Sets the specified value for the element at the given index.

Signature
public Void set(Integer index, Object listElement)
Parameters

*index*
  Type: Integer
  The index of the list element to set.

*listElement*
  Type: Object
  The value of the list element to set.

Return Value
Type: Void

Usage
To set an element of a one-dimensional list of primitives or sObjects, you can also follow the name of the list with the element’s index position in square brackets.

Example

```java
List<Integer> myList = new Integer[6];
myList.set(0, 47);
myList.set(1, 52);
system.assertEquals(52, myList.get(1));

List<String> colors = new String[3];
colors[0] = 'Red';
colors[1] = 'Blue';
colors[2] = 'Green';
```

`size()`

Returns the number of elements in the list.

Signature

```java
public Integer size()
```

Return Value
Type: Integer

Example

```java
List<Integer> myList = new List<Integer>();
Integer size = myList.size();
system.assertEquals(0, size);

List<Integer> myList2 = new Integer[6];
```
sort()
Sorts the items in the list in ascending order.

Signature
public Void sort()

Return Value
Type: Void

Usage
Using this method, you can sort primitive types, SelectOption elements, and sObjects (standard objects and custom objects). For more information on the sort order used for sObjects, see Sorting Lists of sObjects. You can sort custom types (your Apex classes) if they implement the Comparable interface. Alternatively, a class implementing the Comparator interface can be passed as a parameter to the List.sort method.

When you use sort() methods on List<Id>s that contain both 15-character and 18-character IDs, IDs for the same record sort together in API version 35.0 and later.

Example
In the following example, the list has three elements. When the list is sorted, the first element is null because it has no value assigned. The second element and third element have values of 5 and 10.

```java
List<Integer> q1 = new Integer[3];

// Assign values to the first two elements.
q1[0] = 10;
q1[1] = 5;

q1.sort();

// Verify sorted list. Elements are sorted in nulls-first order: null, 5, and 10
system.assertEquals(null, q1.get(0));
system.assertEquals(5, q1.get(1));
system.assertEquals(10, q1.get(2));
```
**toString()**

Returns the string representation of the list.

**Signature**

```java
public String toString()
```

**Return Value**

Type: String

**Usage**

When used in cyclic references, the output is truncated to prevent infinite recursion. When used with large collections, the output is truncated to avoid exceeding total heap size and maximum CPU time.

- Up to 10 items per collection are included in the output, followed by an ellipsis (...).
- If the same object is included multiple times in a collection, it’s shown in the output only once; subsequent references are shown as (already output).

**Location Class**

Contains methods for accessing the component fields of geolocation compound fields.

**Namespace**

`system`

**Usage**

Each of these methods is also equivalent to a read-only property. For each getter method you can access the property using dot notation. For example, `myLocation.getLatitude()` is equivalent to `myLocation.latitude`.

You can’t use dot notation to access compound fields’ subfields directly on the parent field. Instead, assign the parent field to a variable of type `Location`, and then access its components.

```java
Location loc = myAccount.MyLocation__c;
Double lat = loc.latitude;
```

**Important:** “Location” in Salesforce can also refer to the Location standard object. When referencing the Location object in your Apex code, always use `Schema.Location` instead of `Location` to prevent confusion with the standard Location compound field. If referencing both the location object and the Location field in the same snippet, you can differentiate between the two by using `System.Location` for the field and `Schema.Location` for the object.

**Example**

```java
// Select and access the Location field. MyLocation__c is the name of a geolocation field on Account.
Account[] records = [SELECT id, MyLocation__c FROM Account LIMIT 10];
```
for (Account acct : records) {
    Location loc = acct.MyLocation__c;
    Double lat = loc.latitude;
    Double lon = loc.longitude;
}

// Instantiate new Location objects and compute the distance between them in different ways.
Location loc1 = Location.newInstance(28.635308,77.22496);
Location loc2 = Location.newInstance(37.7749295,-122.4194155);
Double dist = Location.getDistance(loc1, loc2, 'mi');
Double dist2 = loc1.getDistance(loc2, 'mi');

IN THIS SECTION:
Location Methods

The following are methods for Location.

IN THIS SECTION:
getDistance(toLocation, unit)
Calculates the distance between this location and the specified location, using an approximation of the haversine formula and the specified unit.

getDistance(firstLocation, secondLocation, unit)
Calculates the distance between the two specified locations, using an approximation of the haversine formula and the specified unit.

getLatitude()
Returns the latitude field of this location.

getLongitude()
Returns the longitude field of this location.

newInstance(latitude, longitude)
Creates an instance of the Location class, with the specified latitude and longitude.

getDistance(toLocation, unit)
Calculates the distance between this location and the specified location, using an approximation of the haversine formula and the specified unit.

Signature
public Double getDistance(Location toLocation, String unit)

Parameters
toLocation
Type: Location
The Location to which you want to calculate the distance from the current Location.

unit
Type: String
The distance unit you want to use: mi or km.

Return Value
Type: Double

gDistance(firstLocation, secondLocation, unit)
Calculates the distance between the two specified locations, using an approximation of the haversine formula and the specified unit.

Signature
public static Double getDistance(Location firstLocation, Location secondLocation, String unit)

Parameters
firstLocation
Type: Location
The first of two locations used to calculate distance.

secondLocation
Type: Location
The second of two locations used to calculate distance.

unit
Type: String
The distance unit you want to use: mi or km.

Return Value
Type: Double

gLatitude()
Returns the latitude field of this location.

Signature
public Double getLatitude()

Return Value
Type: Double
**getLongitude()**

Returns the longitude field of this location.

**Signature**

```java
public Double getLongitude()
```

**Return Value**

Type: Double

**newInstance(latitude, longitude)**

Creates an instance of the `Location` class, with the specified latitude and longitude.

**Signature**

```java
public static Location newInstance(Decimal latitude, Decimal longitude)
```

**Parameters**

- `latitude`
  Type: Decimal
- `longitude`
  Type: Decimal

**Return Value**

Type: Location

---

**LoggingLevel Enum**

Specifies the logging level for the `System.debug` method.

**Enum Values**

The following are the values of the `System.LoggingLevel` enum, listed from the lowest to the highest levels. The level is cumulative, that is, if you select FINE, the log also includes all events logged at the DEBUG, INFO, WARN, and ERROR levels.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NONE</td>
<td>No logging.</td>
</tr>
<tr>
<td>ERROR</td>
<td>Error and exception logging.</td>
</tr>
<tr>
<td>WARN</td>
<td>Warning logging.</td>
</tr>
<tr>
<td>INFO</td>
<td>Informational logging.</td>
</tr>
<tr>
<td>DEBUG</td>
<td>User-specified debug logging.</td>
</tr>
</tbody>
</table>
### Usage

Log levels are cumulative. For example, if the lowest level, ERROR, is specified for Apex code, only `System.debug` methods with the log level of ERROR are logged. If the next log level, WARN, is specified, `System.debug` methods specified with either ERROR or WARN levels are logged.

In this example, if the log level is set to ERROR, the string `MsgTxt` isn't written to the debug log because the `debug` method has a level of INFO.

```java
System.debug(logginglevel.INFO, 'MsgTxt');
```

For more information on log levels, see [Debug Log Levels](https://help.salesforce.com) in Salesforce Help.

## Long Class

Contains methods for the Long primitive data type.

### Namespace

`System`

### Usage

For more information on Long, see [Long Data Type](https://help.salesforce.com).

### Long Methods

The following are methods for `Long`.

**IN THIS SECTION:**

- `format()`: Returns the String format for this Long using the locale of the context user.
- `intValue()`: Returns the Integer value for this Long.
- `valueOf(stringToLong)`: Returns a Long that contains the value of the specified String. As in Java, the string is interpreted as representing a signed decimal Long.
**format()**
Returns the String format for this Long using the locale of the context user.

**Signature**

```java
public String format()
```

**Return Value**
Type: String

**Example**

```java
Long myLong = 4271990;
system.assertEquals('4,271,990', myLong.format());
```

**intValue()**
Returns the Integer value for this Long.

**Signature**

```java
public Integer intValue()
```

**Return Value**
Type: Integer

**Example**

```java
Long myLong = 7191991;
Integer value = myLong.intValue();
system.assertEquals(7191991, myLong.intValue());
```

**valueOf(stringToLong)**
Returns a Long that contains the value of the specified String. As in Java, the string is interpreted as representing a signed decimal Long.

**Signature**

```java
public static Long valueOf(String stringToLong)
```

**Parameters**

- **stringToLong**
  Type: String

**Return Value**
Type: Long
Example

```java
Long l1 = Long.valueOf('123456789');
```

**Map Class**

Contains methods for the Map collection type.

**Namespace**

*System*

**Usage**

The Map methods are all instance methods, that is, they operate on a particular instance of a map. The following are the instance methods for maps.

**Note:**

- Map keys and values can be of any data type—primitive types, collections, sObjects, user-defined types, and built-in Apex types.
- Uniqueness of map keys of user-defined types is determined by the `equals` and `hashCode` methods, which you provide in your classes. Uniqueness of keys of all other non-primitive types, such as sObject keys, is determined by comparing the objects’ field values.
- Map keys of type String are case-sensitive. Two keys that differ only by the case are considered unique and have corresponding distinct Map entries. Subsequently, the Map methods, including `put`, `get`, `containsKey`, and `remove` treat these keys as distinct.

For more information on maps, see Maps.

**IN THIS SECTION:**

- Map Constructors
- Map Methods

**Map Constructors**

The following are constructors for Map.

**IN THIS SECTION:**

- `Map<T1,T2>()` Creates a new instance of the `Map` class. T1 is the data type of the keys and T2 is the data type of the values.
- `Map<T1,T2>(mapToCopy)` Creates a new instance of the `Map` class and initializes it by copying the entries from the specified map. T1 is the data type of the keys and T2 is the data type of the values.
- `Map<ID,sObject>(recordList)` Creates a new instance of the `Map` class and populates it with the passed-in list of sObject records. The keys are populated with the sObject IDs and the values are the sObjects.
**Map<T1,T2>()**

Creates a new instance of the `Map` class. T1 is the data type of the keys and T2 is the data type of the values.

**Signature**

```java
public Map<T1,T2>()
```

**Example**

```java
Map<Integer, String> m1 = new Map<Integer, String>();
m1.put(1, 'First item');
m1.put(2, 'Second item');
```

**Map<T1,T2>(mapToCopy)**

Creates a new instance of the `Map` class and initializes it by copying the entries from the specified map. T1 is the data type of the keys and T2 is the data type of the values.

**Signature**

```java
public Map<T1,T2>(Map<T1,T2> mapToCopy)
```

**Parameters**

`mapToCopy`

Type: `Map<T1, T2>`

The map to initialize this map with. T1 is the data type of the keys and T2 is the data type of the values. All map keys and values are copied to this map.

**Example**

```java
Map<Integer, String> m1 = new Map<Integer, String>();
m1.put(1, 'First item');
m1.put(2, 'Second item');
Map<Integer, String> m2 = new Map<Integer, String>(m1);
// The map elements of m2 are copied from m1
System.debug(m2);
```

**Map<ID,sObject>(recordList)**

Creates a new instance of the `Map` class and populates it with the passed-in list of `sObject` records. The keys are populated with the `sObject` IDs and the values are the `sObjects`.

**Signature**

```java
public Map<ID,sObject>(List<sObject> recordList)
```
Parameters

recordList

Type: List<sObject>

The list of sObjects to populate the map with.

Example

List<Account> ls = [select Id, Name from Account];
Map<Id, Account> m = new Map<Id, Account>(ls);

Map Methods

The following are methods for Map. All are instance methods.

IN THIS SECTION:

clear()
Removes all of the key-value mappings from the map.

copy()
Makes a duplicate copy of the map.

containsKey(key)
Returns true if the map contains a mapping for the specified key.

depthClone()
Makes a duplicate copy of a map, including sObject records if this is a map with sObject record values.

equals(map2)
Compares this map with the specified map and returns true if both maps are equal; otherwise, returns false.

get(key)
Returns the value to which the specified key is mapped, or null if the map contains no value for this key.

getSObjectType()
Returns the token of the sObject type that makes up the map values.

hashCode()
Returns the hashcode corresponding to this map.

isEmpty()
Returns true if the map has zero key-value pairs.

keySet()
Returns a set that contains all of the keys in the map.

put(key, value)
Associates the specified value with the specified key in the map.

putAll(fromMap)
Copies all of the mappings from the specified map to the original map.

putAll(sobjectArray)
Adds the list of sObject records to a map declared as Map<ID, sObject> or Map<String, sObject>. 
remove(key)
Removes the mapping for the specified key from the map, if present, and returns the corresponding value.

size()
Returns the number of key-value pairs in the map.

toString()
Returns the string representation of the map.

values()
Returns a list that contains all the values in the map.

clear()
Removes all of the key-value mappings from the map.

**Signature**

```
public Void clear()
```

**Return Value**

Type: Void

clone()
Makes a duplicate copy of the map.

**Signature**

```
public Map<Object, Object> clone()
```

**Return Value**

Type: Map (of same type)

**Usage**

If this is a map with sObject record values, the duplicate map will only be a shallow copy of the map. That is, the duplicate will have references to each sObject record, but the records themselves are not duplicated. For example:

To also copy the sObject records, you must use the `deepClone` method.

**Example**

```java
Account a = new Account(
    Name='Acme',
    BillingCity='New York');

Map<Integer, Account> map1 = new Map<Integer, Account> {};
map1.put(1, a);

Map<Integer, Account> map2 = map1.clone();
```
map1.get(1).BillingCity = 'San Francisco';

System.assertEquals('San Francisco',
    map1.get(1).BillingCity);

System.assertEquals('San Francisco',
    map2.get(1).BillingCity);

**containsKey(key)**

Returns true if the map contains a mapping for the specified key.

**Signature**

```java
public Boolean containsKey(Object key)
```

**Parameters**

- **key**
  Type: Object

**Return Value**

Type: Boolean

**Usage**

If the key is a string, the key value is case-sensitive.

**Example**

```java
Map<String, String> colorCodes = new Map<String, String>();

colorCodes.put('Red', 'FF0000');
colorCodes.put('Blue', '0000A0');

Boolean contains = colorCodes.containsKey('Blue');
System.assertEquals(true, contains);
```

**deepClone()**

Makes a duplicate copy of a map, including sObject records if this is a map with sObject record values.

**Signature**

```java
public Map<Object, Object> deepClone()
```
Return Value
Type: Map (of the same type)

Usage
To make a shallow copy of a map without duplicating the sObject records it contains, use the clone() method.

Example

```java
Account a = new Account(
    Name='Acme',
    BillingCity='New York');

Map<Integer, Account> map1 = new Map<Integer, Account>{};
map1.put(1, a);

Map<Integer, Account> map2 = map1.deepClone();

// Update the first entry of map1
map1.get(1).BillingCity = 'San Francisco';
// Verify that the BillingCity is updated in map1 but not in map2
System.assertEquals('San Francisco', map1.get(1).BillingCity);
System.assertEquals('New York', map2.get(1).BillingCity);
```

equals(map2)
Compares this map with the specified map and returns true if both maps are equal; otherwise, returns false.

Signature
public Boolean equals(Map map2)

Parameters
map2
Type: Map
The map2 argument is the map to compare this map with.

Return Value
Type: Boolean

Usage
Two maps are equal if their key/value pairs are identical, regardless of the order of those pairs. The == operator is used to compare the map keys and values.

The == operator is equivalent to calling the equals method, so you can call map1.equals(map2); instead of map1 == map2;.
**get (key)**

Returns the value to which the specified key is mapped, or `null` if the map contains no value for this key.

**Signature**

```java
public Object get(Object key)
```

**Parameters**

`key`

Type: Object

**Return Value**

Type: Object

**Usage**

If the key is a string, the key value is case-sensitive.

**Example**

```java
Map<String, String> colorCodes = new Map<String, String>();
colorCodes.put('Red', 'FF0000');
colorCodes.put('Blue', '0000A0');
String code = colorCodes.get('Blue');
System.assertEquals('0000A0', code);
// The following is not a color in the map
String code2 = colorCodes.get('Magenta');
System.assertEquals(null, code2);
```

**getSObjectType ()**

Returns the token of the sObject type that makes up the map values.

**Signature**

```java
public Schema.SObjectType getSObjectType()
```

**Return Value**

Type: `Schema.SObjectType`

**Usage**

Use this method with describe information, to determine if a map contains sObjects of a particular type.
Note that this method can only be used with maps that have sObject values.
For more information, see Understanding Apex Describe Information.

Example

```java
// Create a generic sObject variable.
SObject sObj = Database.query('SELECT Id FROM Account LIMIT 1');

// Verify if that sObject variable is an Account token.
System.assertEquals(Account.sObjectType, sObj.getSObjectType());

// Create a map of generic sObjects
Map<Integer, Account> m = new Map<Integer, Account>();

// Verify if the map contains Account tokens.
System.assertEquals(Account.sObjectType, m.getSObjectType());
```

**hashCode()**

Returns the hashcode corresponding to this map.

**Signature**

```
public Integer hashCode()
```

**Return Value**

Type: Integer

**isEmpty()**

Returns true if the map has zero key-value pairs.

**Signature**

```
public Boolean isEmpty()
```

**Return Value**

Type: Boolean

Example

```java
Map<String, String> colorCodes = new Map<String, String>();
Boolean empty = colorCodes.isEmpty();
System.assertEquals(true, empty);
```
**keySet()**

Returns a set that contains all of the keys in the map.

**Signature**

```java
public Set<Object> keySet()
```

**Return Value**

Type: Set (of key type)

**Example**

```java
Map<String, String> colorCodes = new Map<String, String>();
colorCodes.put('Red', 'FF0000');
colorCodes.put('Blue', '0000A0');
Set<String> colorSet = new Set<String>();
colorSet = colorCodes.keySet();
```

**put(key, value)**

Associates the specified value with the specified key in the map.

**Signature**

```java
public Object put(Object key, Object value)
```

**Parameters**

- `key`
  - Type: Object
- `value`
  - Type: Object

**Return Value**

Type: Object

**Usage**

If the map previously contained a mapping for this key, the old value is returned by the method and then replaced.

If the key is a string, the key value is case-sensitive.

**Example**

```java
Map<String, String> colorCodes = new Map<String, String>();
```
colorCodes.put('Red', 'ff0000');
colorCodes.put('Red', '#FF0000');
// Red is now #FF0000

**putAll(fromMap)**

Copies all of the mappings from the specified map to the original map.

**Signature**

```java
public Void putAll(Map fromMap)
```

**Parameters**

`fromMap`

Type: Map

**Return Value**

Type: Void

**Usage**

The new mappings from `fromMap` replace any mappings that the original map had.

**Example**

```java
Map<String, String> map1 = new Map<String, String>();
map1.put('Red', 'FF0000');
Map<String, String> map2 = new Map<String, String>();
map2.put('Blue', '0000FF');
// Add map1 entries to map2
map2.putAll(map1);
System.assertEquals(2, map2.size());
```

**putAll(subjectArray)**

Adds the list of sObject records to a map declared as Map<ID, sObject> or Map<String, sObject>.

**Signature**

```java
public Void putAll(sObject[] subjectArray)
```

**Parameters**

`subjectArray`

Type: sObject[]

**Return Value**

Type: Void
Usage
This method is similar to calling the Map constructor with the same input.

Example

```java
List<Account> accts = new List<Account>();
accts.add(new Account(Name='Account1'));
accts.add(new Account(Name='Account2'));
// Insert accounts so their IDs are populated.
insert accts;
Map<Id, Account> m = new Map<Id, Account>();
// Add all the records to the map.
m.putAll(accts);
System.assertEquals(2, m.size());
```

**remove(key)**

Removes the mapping for the specified key from the map, if present, and returns the corresponding value.

**Signature**

```
public Object remove(Key key)
```

**Parameters**

- `key`
  
  Type: Key

**Return Value**

Type: Object

**Usage**

If the key is a string, the key value is case-sensitive.

**Example**

```java
Map<String, String> colorCodes = new Map<String, String>();

colorCodes.put('Red', 'FF0000');
colorCodes.put('Blue', '0000A0');

String myColor = colorCodes.remove('Blue');
String code2 = colorCodes.get('Blue');
System.assertEquals(null, code2);
```

**size()**

Returns the number of key-value pairs in the map.
Signature

public Integer size()

Return Value

Type: Integer

Example

Map<String, String> colorCodes = new Map<String, String>();

colorCodes.put('Red', 'FF0000');
colorCodes.put('Blue', '0000A0');

Integer mSize = colorCodes.size();

system.assertEquals(2, mSize);

toString()

Returns the string representation of the map.

Signature

public String toString()

Return Value

Type: String

Usage

When used in cyclic references, the output is truncated to prevent infinite recursion. When used with large collections, the output is truncated to avoid exceeding total heap size and maximum CPU time.

- Up to 10 items per collection are included in the output, followed by an ellipsis (...).
- If the same object is included multiple times in a collection, it’s shown in the output only once; subsequent references are shown as (already output).

values()

Returns a list that contains all the values in the map.

Signature

public List<Object> values()

Return Value

Type: List<Object>
Usage

The order of map elements is deterministic. You can rely on the order being the same in each subsequent execution of the same code. For example, suppose the `values()` method returns a list containing `value1` and index 0 and `value2` and index 1. Subsequent runs of the same code result in those values being returned in the same order.

Example

```java
Map<String, String> colorCodes = new Map<String, String>();
colorCodes.put('Red', 'FF0000');
colorCodes.put('Blue', '0000A0');

List<String> colors = new List<String>();
colors = colorCodes.values();
```

Matcher Class

Matchers use Patterns to perform match operations on a character string.

Namespace

System

Matcher Methods

The following are methods for `Matcher`.

IN THIS SECTION:

- `end()`: Returns the position after the last matched character.
- `end(groupIndex)`: Returns the position after the last character of the subsequence captured by the group index during the previous match operation. If the match was successful but the group itself did not match anything, this method returns -1.
- `find()`: Attempts to find the next subsequence of the input sequence that matches the pattern. This method returns true if a subsequence of the input sequence matches this Matcher object’s pattern.
- `find(group)`: Resets the Matcher object and then tries to find the next subsequence of the input sequence that matches the pattern. This method returns `true` if a subsequence of the input sequence matches this Matcher object’s pattern.
- `group()`: Returns the input subsequence returned by the previous match.
- `group(groupIndex)`: Returns the input subsequence captured by the specified group index during the previous match operation. If the match was successful but the specified group failed to match any part of the input sequence, `null` is returned.
groupCount()
Returns the number of capturing groups in this Matching object’s pattern. Group zero denotes the entire pattern and is not included in this count.

hasAnchoringBounds()
Returns true if the Matcher object has anchoring bounds, false otherwise. By default, a Matcher object uses anchoring bounds regions.

hasTransparentBounds()
Returns true if the Matcher object has transparent bounds, false if it uses opaque bounds. By default, a Matcher object uses opaque region boundaries.

hitEnd()
Returns true if the end of input was found by the search engine in the last match operation performed by this Matcher object. When this method returns true, it is possible that more input would have changed the result of the last search.

lookingAt()
Attempts to match the input sequence, starting at the beginning of the region, against the pattern.

matches()
Attempts to match the entire region against the pattern.

pattern()
Returns the Pattern object from which this Matcher object was created.

quoteReplacement(inputString)
Returns a literal replacement string for the specified string inputString. The characters in the returned string match the sequence of characters in inputString.

region(start, end)
Sets the limits of this Matcher object’s region. The region is the part of the input sequence that is searched to find a match.

regionEnd()
Returns the end index (exclusive) of this Matcher object’s region.

regionStart()
Returns the start index (inclusive) of this Matcher object’s region.

replaceAll(replacementString)
Replaces every subsequence of the input sequence that matches the pattern with the replacement string.

replaceFirst(replacementString)
Replaces the first subsequence of the input sequence that matches the pattern with the replacement string.

requireEnd()
Returns true if more input could change a positive match into a negative one.

reset()
Resets this Matcher object. Resetting a Matcher object discards all of its explicit state information.

reset(inputSequence)
Resets this Matcher object with the new input sequence. Resetting a Matcher object discards all of its explicit state information.

start()
Returns the start index of the first character of the previous match.
start(groupIndex)
Returns the start index of the subsequence captured by the group specified by the group index during the previous match operation. Captured groups are indexed from left to right, starting at one. Group zero denotes the entire pattern, so the expression m.start(0) is equivalent to m.start().

useAnchoringBounds(anchoringBounds)
Sets the anchoring bounds of the region for the Matcher object. By default, a Matcher object uses anchoring bounds regions.

usePattern(pattern)
Changes the Pattern object that the Matcher object uses to find matches. This method causes the Matcher object to lose information about the groups of the last match that occurred. The Matcher object's position in the input is maintained.

useTransparentBounds(transparentBounds)
Sets the transparency bounds for this Matcher object. By default, a Matcher object uses anchoring bounds regions.

end()
Returns the position after the last matched character.

Signature
public Integer end()

Return Value
Type: Integer

end(groupIndex)
Returns the position after the last character of the subsequence captured by the group index during the previous match operation. If the match was successful but the group itself did not match anything, this method returns -1.

Signature
public Integer end(Integer groupIndex)

Parameters

<table>
<thead>
<tr>
<th>groupIndex</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type: Integer</td>
</tr>
</tbody>
</table>

Return Value
Type: Integer

Usage
Captured groups are indexed from left to right, starting at one. Group zero denotes the entire pattern, so the expression m.end(0) is equivalent to m.end().
See Understanding Capturing Groups.
**find()**
Attempts to find the next subsequence of the input sequence that matches the pattern. This method returns true if a subsequence of the input sequence matches this Matcher object's pattern.

**Signature**
```
public Boolean find()
```

**Return Value**
Type: Boolean

**Usage**
This method starts at the beginning of this Matcher object's region, or, if a previous invocation of the method was successful and the Matcher object has not since been reset, at the first character not matched by the previous match.

If the match succeeds, more information can be obtained using the `start`, `end`, and `group` methods.

For more information, see Using Regions.

**find(group)**
Resets the Matcher object and then tries to find the next subsequence of the input sequence that matches the pattern. This method returns true if a subsequence of the input sequence matches this Matcher object's pattern.

**Signature**
```
public Boolean find(Integer group)
```

**Parameters**
- `group`
  Type: Integer

**Return Value**
Type: Boolean

**Usage**
If the match succeeds, more information can be obtained using the `start`, `end`, and `group` methods.

**group()**
Returns the input subsequence returned by the previous match.

**Signature**
```
public String group()
```
Return Value
Type: String

Usage
Note that some groups, such as `(a*)`, match the empty string. This method returns the empty string when such a group successfully matches the empty string in the input.

group(groupIndex)
Returns the input subsequence captured by the specified group index during the previous match operation. If the match was successful but the specified group failed to match any part of the input sequence, `null` is returned.

Signature
public String group(Integer groupIndex)

Parameters

  groupIndex
    Type: Integer

Return Value
Type: String

Usage
Captured groups are indexed from left to right, starting at one. Group zero denotes the entire pattern, so the expression `m.group(0)` is equivalent to `m.group()`.

Note that some groups, such as `(a*)`, match the empty string. This method returns the empty string when such a group successfully matches the empty string in the input.

See Understanding Capturing Groups.

groupCount()
Returns the number of capturing groups in this Matching object’s pattern. Group zero denotes the entire pattern and is not included in this count.

Signature
public Integer groupCount()

Return Value
Type: Integer

Usage
See Understanding Capturing Groups.
hasAnchoringBounds()
Returns true if the Matcher object has anchoring bounds, false otherwise. By default, a Matcher object uses anchoring bounds regions.

Signature
public Boolean hasAnchoringBounds()

Return Value
Type: Boolean

Usage
If a Matcher object uses anchoring bounds, the boundaries of this Matcher object’s region match start and end of line anchors such as ^ and $.
For more information, see Using Bounds.

hasTransparentBounds()
Returns true if the Matcher object has transparent bounds, false if it uses opaque bounds. By default, a Matcher object uses opaque region boundaries.

Signature
public Boolean hasTransparentBounds()

Return Value
Type: Boolean

Usage
For more information, see Using Bounds.

hitEnd()
Returns true if the end of input was found by the search engine in the last match operation performed by this Matcher object. When this method returns true, it is possible that more input would have changed the result of the last search.

Signature
public Boolean hitEnd()

Return Value
Type: Boolean

lookingAt()
Attempts to match the input sequence, starting at the beginning of the region, against the pattern.
Signature

public Boolean lookingAt()

Return Value

Type: Boolean

Usage

Like the matches method, this method always starts at the beginning of the region; unlike that method, it does not require the entire region be matched.

If the match succeeds, more information can be obtained using the start, end, and group methods.

See Using Regions.

matches()

Attempts to match the entire region against the pattern.

Signature

public Boolean matches()

Return Value

Type: Boolean

Usage

If the match succeeds, more information can be obtained using the start, end, and group methods.

See Using Regions.

pattern()

Returns the Pattern object from which this Matcher object was created.

Signature

public Pattern object pattern()

Return Value

Type: System.Pattern

quoteReplacement(inputString)

Returns a literal replacement string for the specified string inputString. The characters in the returned string match the sequence of characters in inputString.
**Signature**

```java
public static String quoteReplacement(String inputString)
```

**Parameters**

`inputString`
Type: String

**Return Value**

Type: String

**Usage**

Metacharacters (such as $ or ^) and escape sequences in the input string are treated as literal characters with no special meaning.

**region(start, end)**

Sets the limits of this Matcher object’s region. The region is the part of the input sequence that is searched to find a match.

**Signature**

```java
public Matcher object region(Integer start, Integer end)
```

**Parameters**

`start`
Type: Integer

`end`
Type: Integer

**Return Value**

Type: Matcher

**Usage**

This method first resets the Matcher object, then sets the region to start at the index specified by `start` and end at the index specified by `end`.

Depending on the transparency boundaries being used, certain constructs such as anchors may behave differently at or around the boundaries of the region.

See Using Regions and Using Bounds.

**regionEnd()**

Returns the end index (exclusive) of this Matcher object’s region.
Signature

```
public Integer regionEnd()
```

Return Value

Type: Integer

Usage

See Using Regions.

`regionStart()`

Returns the start index (inclusive) of this Matcher object’s region.

Signature

```
public Integer regionStart()
```

Return Value

Type: Integer

Usage

See Using Regions.

`replaceAll(replacementString)`

Replaces every subsequence of the input sequence that matches the pattern with the replacement string.

Signature

```
public String replaceAll(String replacementString)
```

Parameters

- `replacementString`
  Type: String

Return Value

Type: String

Usage

This method first resets the Matcher object, then scans the input sequence looking for matches of the pattern. Characters that are not part of any match are appended directly to the result string; each match is replaced in the result by the replacement string. The replacement string may contain references to captured subsequences.
Note that backslashes (\) and dollar signs ($) in the replacement string may cause the results to be different than if the string was treated as a literal replacement string. Dollar signs may be treated as references to captured subsequences, and backslashes are used to escape literal characters in the replacement string.

Invoking this method changes this Matcher object’s state. If the Matcher object is to be used in further matching operations it should first be reset.

Given the regular expression *a+b*, the input "aabxyzaabxyzabxyzb", and the replacement string "-", an invocation of this method on a Matcher object for that expression would yield the string "-xyz-xyz-xyz-".

**replaceFirst(replacementString)**
Replaces the first subsequence of the input sequence that matches the pattern with the replacement string.

**Signature**
`public String replaceFirst(String replacementString)`

**Parameters**
- `replacementString`
  Type: String

**Return Value**
Type: String

**Usage**
Note that backslashes (\) and dollar signs ($) in the replacement string may cause the results to be different than if the string was treated as a literal replacement string. Dollar signs may be treated as references to captured subsequences, and backslashes are used to escape literal characters in the replacement string.

Invoking this method changes this Matcher object’s state. If the Matcher object is to be used in further matching operations it should first be reset.

Given the regular expression *dog*, the input "zzzdogzzzdogzzz", and the replacement string "cat", an invocation of this method on a Matcher object for that expression would return the string "zzzcatzzzdogzzz".

**requireEnd()**
Returns true if more input could change a positive match into a negative one.

**Signature**
`public Boolean requireEnd()`

**Return Value**
Type: Boolean
Usage
If this method returns true, and a match was found, then more input could cause the match to be lost.
If this method returns false and a match was found, then more input might change the match but the match won’t be lost.
If a match was not found, then requireEnd has no meaning.

reset()
Resets this Matcher object. Resetting a Matcher object discards all of its explicit state information.

Signature
public Matcher object reset()

Return Value
Type: Matcher

Usage
This method does not change whether the Matcher object uses anchoring bounds. You must explicitly use the useAnchoringBounds method to change the anchoring bounds.
For more information, see Using Bounds.

reset(inputSequence)
Resets this Matcher object with the new input sequence. Resetting a Matcher object discards all of its explicit state information.

Signature
public Matcher reset(String inputSequence)

Parameters
inputSequence
Type: String

Return Value
Type: Matcher

start()
Returns the start index of the first character of the previous match.

Signature
public Integer start()
Return Value
Type: Integer

**start(groupIndex)**

Returns the start index of the subsequence captured by the group specified by the group index during the previous match operation. Captured groups are indexed from left to right, starting at one. Group zero denotes the entire pattern, so the expression `m.start(0)` is equivalent to `m.start()`.

**Signature**

```
public Integer start(Integer groupIndex)
```

**Parameters**

- `groupIndex`
  Type: Integer

**Return Value**

Type: Integer

**Usage**

See Understanding Capturing Groups.

**useAnchoringBounds(anchoringBounds)**

Sets the anchoring bounds of the region for the Matcher object. By default, a Matcher object uses anchoring bounds regions.

**Signature**

```
public Matcher object useAnchoringBounds(Boolean anchoringBounds)
```

**Parameters**

- `anchoringBounds`
  Type: Boolean
  If you specify `true`, the Matcher object uses anchoring bounds. If you specify `false`, non-anchoring bounds are used.

**Return Value**

Type: Matcher

**Usage**

If a Matcher object uses anchoring bounds, the boundaries of this Matcher object’s region match start and end of line anchors such as `^` and `$.

For more information, see Using Bounds.
**usePattern (pattern)**
Changes the Pattern object that the Matcher object uses to find matches. This method causes the Matcher object to lose information about the groups of the last match that occurred. The Matcher object’s position in the input is maintained.

**Signature**
```java
public Matcher object usePattern(Pattern pattern)
```

**Parameters**
- `pattern`  
  Type: `System.Pattern`

**Return Value**
Type: `Matcher`

---

**useTransparentBounds (transparentBounds)**
Sets the transparency bounds for this Matcher object. By default, a Matcher object uses anchoring bounds regions.

**Signature**
```java
public Matcher object useTransparentBounds(Boolean transparentBounds)
```

**Parameters**
- `transparentBounds`  
  Type: `Boolean`

  If you specify `true`, the Matcher object uses transparent bounds. If you specify `false`, opaque bounds are used.

**Return Value**
Type: `Matcher`

**Usage**
For more information, see `Using Bounds`.

---

**Math Class**
Contains methods for mathematical operations.

**Namespace**
`System`
Math Fields

The following are fields for Math.

IN THIS SECTION:

E
Returns the mathematical constant e, which is the base of natural logarithms.

PI
Returns the mathematical constant pi, which is the ratio of the circumference of a circle to its diameter.

E
Returns the mathematical constant e, which is the base of natural logarithms.

Signature

public static final Double E

Property Value

Type: Double

PI
Returns the mathematical constant pi, which is the ratio of the circumference of a circle to its diameter.

Signature

public static final Double PI

Property Value

Type: Double

Math Methods

The following are methods for Math. All methods are static.

IN THIS SECTION:

abs(decimalValue)
Returns the absolute value of the specified Decimal.

abs(doubleValue)
Returns the absolute value of the specified Double.

abs(integerValue)
Returns the absolute value of the specified Integer.

abs(longValue)
Returns the absolute value of the specified Long.
acos(decimalAngle)
Returns the arc cosine of an angle, in the range of 0.0 through pi.
acos(doubleAngle)
Returns the arc cosine of an angle, in the range of 0.0 through pi.

asin(decimalAngle)
Returns the arc sine of an angle, in the range of -pi/2 through pi/2.
asin(doubleAngle)
Returns the arc sine of an angle, in the range of -pi/2 through pi/2.

atan(decimalAngle)
Returns the arc tangent of an angle, in the range of -pi/2 through pi/2.
atan(doubleAngle)
Returns the arc tangent of an angle, in the range of -pi/2 through pi/2.

atan2(xCoordinate, yCoordinate)
Converts rectangular coordinates (xCoordinate and yCoordinate) to polar (r and theta). This method computes the phase theta by computing an arc tangent of xCoordinate/yCoordinate in the range of -pi to pi.

ctb(decimalValue)
Returns the cube root of the specified Decimal. The cube root of a negative value is the negative of the cube root of that value's magnitude.
ctb(doubleValue)
Returns the cube root of the specified Double. The cube root of a negative value is the negative of the cube root of that value's magnitude.

ceil(decimalValue)
Returns the smallest (closest to negative infinity) Decimal that is not less than the argument and is equal to a mathematical integer.
ceil(doubleValue)
Returns the smallest (closest to negative infinity) Double that is not less than the argument and is equal to a mathematical integer.

cos(decimalAngle)
Returns the trigonometric cosine of the angle specified by decimalAngle.
cos(doubleAngle)
Returns the trigonometric cosine of the angle specified by doubleAngle.

cosh(decimalAngle)
Returns the hyperbolic cosine of decimalAngle. The hyperbolic cosine of d is defined to be \((e^d + e^{-d})/2\) where e is Euler's number.
cosh(doubleAngle)
Returns the hyperbolic cosine of doubleAngle. The hyperbolic cosine of d is defined to be \((e^d + e^{-d})/2\) where e is Euler's number.

exp(exponentDecimal)
Returns Euler's number e raised to the power of the specified Decimal.
exp(exponentDouble)
Returns Euler's number e raised to the power of the specified Double.
floor(decimalValue)
Returns the largest (closest to positive infinity) Decimal that is not greater than the argument and is equal to a mathematical integer.

floor(doubleValue)
Returns the largest (closest to positive infinity) Double that is not greater than the argument and is equal to a mathematical integer.

log(decimalValue)
Returns the natural logarithm (base $e$) of the specified Decimal.

log(doubleValue)
Returns the natural logarithm (base $e$) of the specified Double.

log10(decimalValue)
Returns the logarithm (base 10) of the specified Decimal.

log10(doubleValue)
Returns the logarithm (base 10) of the specified Double.

max(decimalValue1, decimalValue2)
Returns the larger of the two specified Decimals.

max(doubleValue1, doubleValue2)
Returns the larger of the two specified Doubles.

max(integerValue1, integerValue2)
Returns the larger of the two specified Integers.

max(longValue1, longValue2)
Returns the larger of the two specified Longs.

min(decimalValue1, decimalValue2)
Returns the smaller of the two specified Decimals.

min(doubleValue1, doubleValue2)
Returns the smaller of the two specified Doubles.

min(integerValue1, integerValue2)
Returns the smaller of the two specified Integers.

min(longValue1, longValue2)
Returns the smaller of the two specified Longs.

mod(integerValue1, integerValue2)
Returns the remainder of integerValue1 divided by integerValue2.

mod(longValue1, longValue2)
Returns the remainder of longValue1 divided by longValue2.

pow(doubleValue, exponent)
Returns the value of the first Double raised to the power of exponent.

random()
Returns a positive Double that is greater than or equal to 0.0 and less than 1.0.

rint(decimalValue)
Returns the value that is closest in value to decimalValue and is equal to a mathematical integer.

rint(doubleValue)
Returns the value that is closest in value to doubleValue and is equal to a mathematical integer.
round(doubleValue)
Do not use. This method is deprecated as of the Winter '08 release. Instead, use Math.roundToLong. Returns the closest Integer to the specified Double. If the result is less than -2,147,483,648 or greater than 2,147,483,647, Apex generates an error.

round(decimalValue)
Returns the rounded approximation of this Decimal. The number is rounded to zero decimal places using half-even rounding mode, that is, it rounds towards the “nearest neighbor” unless both neighbors are equidistant, in which case, this mode rounds towards the even neighbor.

roundToLong(decimalValue)
Returns the rounded approximation of this Decimal. The number is rounded to zero decimal places using half-even rounding mode, that is, it rounds towards the “nearest neighbor” unless both neighbors are equidistant, in which case, this mode rounds towards the even neighbor.

roundToLong(doubleValue)
Returns the closest Long to the specified Double.

signum(decimalValue)
Returns the signum function of the specified Decimal, which is 0 if `decimalValue` is 0, 1.0 if `decimalValue` is greater than 0, -1.0 if `decimalValue` is less than 0.

signum(doubleValue)
Returns the signum function of the specified Double, which is 0 if `doubleValue` is 0, 1.0 if `doubleValue` is greater than 0, -1.0 if `doubleValue` is less than 0.

sin(decimalAngle)
Returns the trigonometric sine of the angle specified by `decimalAngle`.

sin(doubleAngle)
Returns the trigonometric sine of the angle specified by `doubleAngle`.

sinh(decimalAngle)
Returns the hyperbolic sine of `decimalAngle`. The hyperbolic sine of `decimalAngle` is defined to be \((e^x - e^{-x})/2\) where \(e\) is Euler's number.

sinh(doubleAngle)
Returns the hyperbolic sine of `doubleAngle`. The hyperbolic sine of `doubleAngle` is defined to be \((e^x - e^{-x})/2\) where \(e\) is Euler's number.

sqrt(decimalValue)
Returns the correctly rounded positive square root of `decimalValue`.

sqrt(doubleValue)
Returns the correctly rounded positive square root of `doubleValue`.

tan(decimalAngle)
Returns the trigonometric tangent of the angle specified by `decimalAngle`.

tan(doubleAngle)
Returns the trigonometric tangent of the angle specified by `doubleAngle`.

tanh(decimalAngle)
Returns the hyperbolic tangent of `decimalAngle`. The hyperbolic tangent of `decimalAngle` is defined to be \((e^x - e^{-x})/(e^x + e^{-x})\) where \(e\) is Euler's number. In other words, it is equivalent to \(\sinh(x)/\cosh(x)\). The absolute value of the exact `tanh` is always less than 1.
tanh(doubleAngle)
Returns the hyperbolic tangent of doubleAngle. The hyperbolic tangent of doubleAngle is defined to be \((e^x - e^{-x})/(e^x + e^{-x})\) where \(e\) is Euler’s number. In other words, it is equivalent to \(\sinh(x)/\cosh(x)\). The absolute value of the exact \(\tanh\) is always less than 1.

abs (decimalValue)
Returns the absolute value of the specified Decimal.

**Signature**

```java
public static Decimal abs(Decimal decimalValue)
```

**Parameters**

decimalValue
Type: Decimal

**Return Value**

Type: Decimal

abs (doubleValue)
Returns the absolute value of the specified Double.

**Signature**

```java
public static Double abs(Double doubleValue)
```

**Parameters**

doubleValue
Type: Double

**Return Value**

Type: Double

abs (integerValue)
Returns the absolute value of the specified Integer.

**Signature**

```java
public static Integer abs(Integer integerValue)
```

**Parameters**

integerValue
Type: Integer
Return Value
Type: Integer

Example

```java
Integer i = -42;
Integer i2 = math.abs(i);
system.assertEquals(i2, 42);
```

`abs(longValue)`
Returns the absolute value of the specified Long.

**Signature**

```java
public static Long abs(Long longValue)
```

**Parameters**

`longValue`  
Type: Long

Return Value  
Type: Long

`acos(decimalAngle)`
Returns the arc cosine of an angle, in the range of 0.0 through \( \pi \).

**Signature**

```java
public static Decimal acos(Decimal decimalAngle)
```

**Parameters**

`decimalAngle`  
Type: Decimal

Return Value  
Type: Decimal

`acos(doubleAngle)`
Returns the arc cosine of an angle, in the range of 0.0 through \( \pi \).

**Signature**

```java
public static Double acos(Double doubleAngle)
```
Parameters

\textit{doubleAngle}

Type: \textit{Double}

Return Value

Type: \textit{Double}

\textbf{asin(decimalAngle)}

Returns the arc sine of an angle, in the range of \(-\pi/2 \text{ through } \pi/2\).

Signature

\texttt{public static Decimal asin(Decimal decimalAngle)}

Parameters

\textit{decimalAngle}

Type: \textit{Decimal}

Return Value

Type: \textit{Decimal}

\textbf{asin(doubleAngle)}

Returns the arc sine of an angle, in the range of \(-\pi/2 \text{ through } \pi/2\).

Signature

\texttt{public static Double asin(Double doubleAngle)}

Parameters

\textit{doubleAngle}

Type: \textit{Double}

Return Value

Type: \textit{Double}

\textbf{atan(decimalAngle)}

Returns the arc tangent of an angle, in the range of \(-\pi/2 \text{ through } \pi/2\).

Signature

\texttt{public static Decimal atan(Decimal decimalAngle)}
Parameters

decimalAngle
   Type: Decimal

Return Value
   Type: Decimal

atan(doubleAngle)
Returns the arc tangent of an angle, in the range of -pi/2 through pi/2.

Signature
public static Double atan(Double doubleAngle)

Parameters

doubleAngle
   Type: Double

Return Value
   Type: Double

atan2(xCoordinate, yCoordinate)
Converts rectangular coordinates (xCoordinate and yCoordinate) to polar (r and theta). This method computes the phase theta by computing an arc tangent of xCoordinate/yCoordinate in the range of -pi to pi.

Signature
public static Decimal atan2(Decimal xCoordinate, Decimal yCoordinate)

Parameters

xCoordinate
   Type: Decimal

yCoordinate
   Type: Decimal

Return Value
   Type: Decimal

atan2(xCoordinate, yCoordinate)
Converts rectangular coordinates (xCoordinate and yCoordinate) to polar (r and theta). This method computes the phase theta by computing an arc tangent of xCoordinate/yCoordinate in the range of -pi to pi.
Signature

```
public static Double atan2(Double xCoordinate, Double yCoordinate)
```

Parameters

- **xCoordinate**
  - Type: Double
- **yCoordinate**
  - Type: Double

Return Value

Type: Double

cbrt(decimalValue)

Returns the cube root of the specified Decimal. The cube root of a negative value is the negative of the cube root of that value’s magnitude.

Signature

```
public static Decimal cbrt(Decimal decimalValue)
```

Parameters

- **decimalValue**
  - Type: Decimal

Return Value

Type: Decimal

cbrt(doubleValue)

Returns the cube root of the specified Double. The cube root of a negative value is the negative of the cube root of that value’s magnitude.

Signature

```
public static Double cbrt(Double doubleValue)
```

Parameters

- **doubleValue**
  - Type: Double

Return Value

Type: Double
**ceil(decimalValue)**

Returns the smallest (closest to negative infinity) Decimal that is not less than the argument and is equal to a mathematical integer.

**Signature**

```java
public static Decimal ceil(Decimal decimalValue)
```

**Parameters**

- `decimalValue`
  - Type: Decimal

**Return Value**

- Type: Decimal

**ceil(doubleValue)**

Returns the smallest (closest to negative infinity) Double that is not less than the argument and is equal to a mathematical integer.

**Signature**

```java
public static Double ceil(Double doubleValue)
```

**Parameters**

- `doubleValue`
  - Type: Double

**Return Value**

- Type: Double

**cos(decimalAngle)**

Returns the trigonometric cosine of the angle specified by `decimalAngle`.

**Signature**

```java
public static Decimal cos(Decimal decimalAngle)
```

**Parameters**

- `decimalAngle`
  - Type: Decimal

**Return Value**

- Type: Decimal
cos(doubleAngle)
Returns the trigonometric cosine of the angle specified by doubleAngle.

Signature
public static Double cos(Double doubleAngle)

Parameters
doubleAngle
   Type: Double

Return Value
Type: Double

cosh(decimalAngle)
Returns the hyperbolic cosine of decimalAngle. The hyperbolic cosine of \( d \) is defined to be \( (e^d + e^{-d})/2 \) where \( e \) is Euler's number.

Signature
public static Decimal cosh(Decimal decimalAngle)

Parameters
decimalAngle
   Type: Decimal

Return Value
Type: Decimal

cosh(doubleAngle)
Returns the hyperbolic cosine of doubleAngle. The hyperbolic cosine of \( d \) is defined to be \( (e^d + e^{-d})/2 \) where \( e \) is Euler's number.

Signature
public static Double cosh(Double doubleAngle)

Parameters
doubleAngle
   Type: Double

Return Value
Type: Double
**exp (exponentDecimal)**
Returns Euler’s number $e$ raised to the power of the specified Decimal.

**Signature**
```
public static Decimal exp(Decimal exponentDecimal)
```

**Parameters**
- `exponentDecimal`
  Type: `Decimal`

**Return Value**
Type: `Decimal`

**exp (exponentDouble)**
Returns Euler’s number $e$ raised to the power of the specified Double.

**Signature**
```
public static Double exp(Double exponentDouble)
```

**Parameters**
- `exponentDouble`
  Type: `Double`

**Return Value**
Type: `Double`

**floor (decimalValue)**
Returns the largest (closest to positive infinity) Decimal that is not greater than the argument and is equal to a mathematical integer.

**Signature**
```
public static Decimal floor(Decimal decimalValue)
```

**Parameters**
- `decimalValue`
  Type: `Decimal`

**Return Value**
Type: `Decimal`
**floor(doubleValue)**
Returns the largest (closest to positive infinity) Double that is not greater than the argument and is equal to a mathematical integer.

**Signature**
```
public static Double floor(Double doubleValue)
```

**Parameters**
-doubleValue
  Type: Double

**Return Value**
Type: Double

**log(decimalValue)**
Returns the natural logarithm (base $e$) of the specified Decimal.

**Signature**
```
public static Decimal log(Decimal decimalValue)
```

**Parameters**
-decimalValue
  Type: Decimal

**Return Value**
Type: Decimal

**log(doubleValue)**
Returns the natural logarithm (base $e$) of the specified Double.

**Signature**
```
public static Double log(Double doubleValue)
```

**Parameters**
-doubleValue
  Type: Double

**Return Value**
Type: Double
log10(decimalValue)
Returns the logarithm (base 10) of the specified Decimal.

Signature
public static Decimal log10(Decimal decimalValue)

Parameters
decimalValue
Type: Decimal

Return Value
Type: Decimal

log10(doubleValue)
Returns the logarithm (base 10) of the specified Double.

Signature
public static Double log10(Double doubleValue)

Parameters
doubleValue
Type: Double

Return Value
Type: Double

max(decimalValue1, decimalValue2)
Returns the larger of the two specified Decimals.

Signature
public static Decimal max(Decimal decimalValue1, Decimal decimalValue2)

Parameters
decimalValue1
Type: Decimal
decimalValue2
Type: Decimal
Return Value
Type: Decimal

Example
```
Decimal larger = math.max(12.3, 156.6);
system.assertEquals(larger, 156.6);
```

max(doubleValue1, doubleValue2)
Returns the larger of the two specified Doubles.

Signature
```
public static Double max(Double doubleValue1, Double doubleValue2)
```

Parameters
```
doubleValue1  
  Type: Double

doubleValue2  
  Type: Double
```

Return Value
Type: Double

max(integerValue1, integerValue2)
Returns the larger of the two specified Integers.

Signature
```
public static Integer max(Integer integerValue1, Integer integerValue2)
```

Parameters
```
integerValue1  
  Type: Integer

integerValue2  
  Type: Integer
```

Return Value
Type: Integer

max(longValue1, longValue2)
Returns the larger of the two specified Longs.
**max(longValue1, longValue2)**

**Parameters**
- `longValue1`
  - Type: `Long`
- `longValue2`
  - Type: `Long`

**Return Value**
- Type: `Long`

**min(decimalValue1, decimalValue2)**

Returns the smaller of the two specified Decimals.

**Signature**

```
public static Decimal min(Decimal decimalValue1, Decimal decimalValue2)
```

**Parameters**
- `decimalValue1`
  - Type: `Decimal`
- `decimalValue2`
  - Type: `Decimal`

**Return Value**
- Type: `Decimal`

**Example**

```java
Decimal smaller = math.min(12.3, 156.6);
system.assertEquals(smaller, 12.3);
```

**min(doubleValue1, doubleValue2)**

Returns the smaller of the two specified Doubles.

**Signature**

```
public static Double min(Double doubleValue1, Double doubleValue2)
```
Parameters

doubleValue1
Type: Double

doubleValue2
Type: Double

Return Value
Type: Double

\texttt{min(integerValue1, integerValue2)}
Returns the smaller of the two specified Integers.

Signature

\texttt{public static Integer min(Integer integerValue1, Integer integerValue2)}

Parameters

integerValue1
Type: Integer

integerValue2
Type: Integer

Return Value
Type: Integer

\texttt{min(longValue1, longValue2)}
Returns the smaller of the two specified Longs.

Signature

\texttt{public static Long min(Long longValue1, Long longValue2)}

Parameters

longValue1
Type: Long

longValue2
Type: Long

Return Value
Type: Long
mod(integerValue1, integerValue2)

Returns the remainder of integerValue1 divided by integerValue2.

Signature

public static Integer mod(Integer integerValue1, Integer integerValue2)

Parameters

integerValue1
Type: Integer

integerValue2
Type: Integer

Return Value

Type: Integer

Example

```java
Integer remainder = math.mod(12, 2);
system.assertEquals(remainder, 0);

Integer remainder2 = math.mod(8, 3);
system.assertEquals(remainder2, 2);
```

mod(longValue1, longValue2)

Returns the remainder of longValue1 divided by longValue2.

Signature

public static Long mod(Long longValue1, Long longValue2)

Parameters

longValue1
Type: Long

longValue2
Type: Long

Return Value

Type: Long

pow(doubleValue, exponent)

Returns the value of the first Double raised to the power of exponent.
public static Double pow(Double doubleValue, Double exponent)

Parameters

doubleValue
  Type: Double

exponent
  Type: Double

Return Value
Type: Double

random()
Returns a positive Double that is greater than or equal to 0.0 and less than 1.0.

signature
public static Double random()

Return Value
Type: Double

rint(decimalValue)
Returns the value that is closest in value to decimalValue and is equal to a mathematical integer.

Signature
public static Decimal rint(Decimal decimalValue)

Parameters

decimalValue
  Type: Decimal

Return Value
Type: Decimal

rint(doubleValue)
Returns the value that is closest in value to doubleValue and is equal to a mathematical integer.

Signature
public static Double rint(Double doubleValue)
Parameters

doubleValue
Type: Double

Return Value
Type: Double

**round (doubleValue)**
Do not use. This method is deprecated as of the Winter '08 release. Instead, use Math.roundToLong. Returns the closest Integer to the specified Double. If the result is less than -2,147,483,648 or greater than 2,147,483,647, Apex generates an error.

Signature

```
public static Integer round(Double doubleValue)
```

Parameters

doubleValue
Type: Double

Return Value
Type: Integer

**round (decimalValue)**
Returns the rounded approximation of this Decimal. The number is rounded to zero decimal places using half-even rounding mode, that is, it rounds towards the “nearest neighbor” unless both neighbors are equidistant, in which case, this mode rounds towards the even neighbor.

Signature

```
public static Integer round(Decimal decimalValue)
```

Parameters

decimalValue
Type: Decimal

Return Value
Type: Integer

Usage
Note that this rounding mode statistically minimizes cumulative error when applied repeatedly over a sequence of calculations.
**Example**

```java
Decimal d1 = 4.5;
Integer i1 = Math.round(d1);
System.assertEquals(4, i1);

Decimal d2 = 5.5;
Integer i2 = Math.round(d2);
System.assertEquals(6, i2);
```

**roundToLong(decimalValue)**

Returns the rounded approximation of this Decimal. The number is rounded to zero decimal places using half-even rounding mode, that is, it rounds towards the "nearest neighbor" unless both neighbors are equidistant, in which case, this mode rounds towards the even neighbor.

**Signature**

```java
public static Long roundToLong(Decimal decimalValue)
```

**Parameters**

decimalValue

Type: Decimal

**Return Value**

Type: Long

**Usage**

Note that this rounding mode statistically minimizes cumulative error when applied repeatedly over a sequence of calculations.

**Example**

```java
Decimal d1 = 4.5;
Long i1 = Math.roundToLong(d1);
System.assertEquals(4, i1);

Decimal d2 = 5.5;
Long i2 = Math.roundToLong(d2);
System.assertEquals(6, i2);
```

**roundToLong(doubleValue)**

Returns the closest Long to the specified Double.

**Signature**

```java
public static Long roundToLong(Double doubleValue)
```
Parameters

doubleValue
   Type: Double

Return Value
Type: Long

**signum(decimalValue)**

Returns the signum function of the specified Decimal, which is 0 if `decimalValue` is 0, 1.0 if `decimalValue` is greater than 0, -1.0 if `decimalValue` is less than 0.

Signature

`public static Decimal signum(Decimal decimalValue)`

Parameters

decimalValue
   Type: Decimal

Return Value
Type: Decimal

**signum(doubleValue)**

Returns the signum function of the specified Double, which is 0 if `doubleValue` is 0, 1.0 if `doubleValue` is greater than 0, -1.0 if `doubleValue` is less than 0.

Signature

`public static Double signum(Double doubleValue)`

Parameters

doubleValue
   Type: Double

Return Value
Type: Double

**sin(decimalAngle)**

Returns the trigonometric sine of the angle specified by `decimalAngle`. 
Signature
public static Decimal sin(Decimal decimalAngle)

Parameters
decimalAngle
  Type: Decimal

Return Value
Type: Decimal

\textbf{sin(doubleAngle)}

Returns the trigonometric sine of the angle specified by doubleAngle.

Signature
public static Double sin(Double doubleAngle)

Parameters
doubleAngle
  Type: Double

Return Value
Type: Double

\textbf{sinh(decimalAngle)}

Returns the hyperbolic sine of decimalAngle. The hyperbolic sine of decimalAngle is defined to be \((e^x - e^{-x})/2\) where \(e\) is Euler's number.

Signature
public static Decimal sinh(Decimal decimalAngle)

Parameters
decimalAngle
  Type: Decimal

Return Value
Type: Decimal
\textbf{sinh(\textit{doubleAngle})}

Returns the hyperbolic sine of \textit{doubleAngle}. The hyperbolic sine of \textit{doubleAngle} is defined to be \((e^x - e^{-x})/2\) where \(e\) is Euler's number.

**Signature**

\texttt{public static Double sinh(Double doubleAngle)}

**Parameters**

- \textit{doubleAngle}
  - Type: Double

**Return Value**

- Type: Double

\textbf{sqrt(\textit{decimalValue})}

Returns the correctly rounded positive square root of \textit{decimalValue}.

**Signature**

\texttt{public static Decimal sqrt(Decimal decimalValue)}

**Parameters**

- \textit{decimalValue}
  - Type: Decimal

**Return Value**

- Type: Decimal

\textbf{sqrt(\textit{doubleValue})}

Returns the correctly rounded positive square root of \textit{doubleValue}.

**Signature**

\texttt{public static Double sqrt(Double doubleValue)}

**Parameters**

- \textit{doubleValue}
  - Type: Double

**Return Value**

- Type: Double
**tan (decimalAngle)**

Returns the trigonometric tangent of the angle specified by `decimalAngle`.

**Signature**

```java
public static Decimal tan(Decimal decimalAngle)
```

**Parameters**

- `decimalAngle`
  Type: `Decimal`

**Return Value**

Type: `Decimal`

**tan (doubleAngle)**

Returns the trigonometric tangent of the angle specified by `doubleAngle`.

**Signature**

```java
public static Double tan(Double doubleAngle)
```

**Parameters**

- `doubleAngle`
  Type: `Double`

**Return Value**

Type: `Double`

**tanh (decimalAngle)**

Returns the hyperbolic tangent of `decimalAngle`. The hyperbolic tangent of `decimalAngle` is defined to be `(e^x - e^{-x})/(e^x + e^{-x})` where `e` is Euler's number. In other words, it is equivalent to `sinh (x) / cosh (x)`. The absolute value of the exact `tanh` is always less than 1.

**Signature**

```java
public static Decimal tanh(Decimal decimalAngle)
```

**Parameters**

- `decimalAngle`
  Type: `Decimal`
Return Value
Type: Decimal

tanh(doubleAngle)
Returns the hyperbolic tangent of doubleAngle. The hyperbolic tangent of doubleAngle is defined to be \((e^x - e^{-x})/(e^x + e^{-x})\) where \(e\) is Euler's number. In other words, it is equivalent to \(\sinh(x)/\cosh(x)\). The absolute value of the exact \(\tanh\) is always less than 1.

Signature
public static Double tanh(Double doubleAngle)

Parameters
doubleAngle
Type: Double

Return Value
Type: Double

Messaging Class
Contains messaging methods used when sending a single or mass email.

Namespace
System

Messaging Methods
The following are methods for Messaging. All are instance methods.

IN THIS SECTION:

extractInboundEmail(source, includeForwardedAttachments)
Use this method in your email service code to control how to parse and process forwarded or attached emails. Returns an instance of Messaging.InboundEmail from a stream of data that is in RFC822 format. The data stream can be a forwarded email in an attachment to an existing InboundEmail, or a stream from another source.

reserveMassEmailCapacity(amountReserved)
Reserves email capacity to send mass email to the specified number of email addresses, after the current transaction commits.

reserveSingleEmailCapacity(amountReserved)
Reserves email capacity to send single email to the specified number of email addresses, after the current transaction commits.
sendEmail(emails, allOrNothing)
Sends the list of emails instantiated with either SingleEmailMessage or MassEmailMessage and returns a list of SendEmailResult objects. When org preferences are set to save EmailMessage objects and a trigger is defined for EmailMessage objects, the trigger is fired for each SingleEmailMessage individually. The sendEmail method can be called 10 times per Apex transaction and each method invocation can include up to 100 "To", 25 "Cc", and 25 "Bcc" recipients.

sendEmailMessage(emailMessageIds, allOrNothing)
Sends draft email messages as defined by the specified email message IDs and returns a list of SendEmailResult objects.

renderEmailTemplate(whoId, whatId, bodies)
Replaces merge fields in text bodies of email templates with values from Salesforce records. Returns an array of RenderEmailTemplateBodyResult objects, each of which corresponds to an element in the supplied array of text bodies. Each RenderEmailTemplateBodyResult provides a success or failure indication, along with either an error code or the rendered text.

renderStoredEmailTemplate(templateId, whoId, whatId)
Renders a text, custom, HTML, or Visualforce email template that exists in the database into an instance of Messaging.SingleEmailMessage. Includes all attachment content in the returned email message.

renderStoredEmailTemplate(templateId, whoId, whatId, attachmentRetrievalOption)
Renders a text, custom, HTML, or Visualforce email template that exists in the database into an instance of Messaging.SingleEmailMessage. Provides options for including attachment metadata only, attachment metadata and content, or excluding attachments.

renderStoredEmailTemplate(templateId, whoId, whatId, attachmentRetrievalOption, updateEmailTemplateUsage)
Renders a text, custom, HTML, or Visualforce email template that exists in the database into an instance of Messaging.SingleEmailMessage. Provides options for including attachment metadata only, attachment metadata and content, or excluding attachments.

extractInboundEmail(source, includeForwardedAttachments)
Use this method in your email service code to control how to parse and process forwarded or attached emails. Returns an instance of Messaging.InboundEmail from a stream of data that is in RFC822 format. The data stream can be a forwarded email in an attachment to an existing InboundEmail, or a stream from another source.

Signature
public static Messaging.InboundEmail extractInboundEmail(Object source, Boolean includeForwardedAttachments)

Parameters

source
Type: Object
An instance of Messaging.InboundEmail.BinaryAttachment whose MimeTypeSubtype is message/rfc822 or a Blob. If source is a Blob, then supply a byte array in RFC822 format.

includeForwardedAttachments
Type: Boolean
This parameter controls how attachments to embedded or forwarded emails are handled. Set to true to provide all attachments, even attachments in embedded emails in the binaryAttachments and textAttachments properties of the returned value. Set to false to provide only the attachments that are at the top level of the source email.
Return Value
Type: Messaging.InboundEmail

reserveMassEmailCapacity(amountReserved)
Reserves email capacity to send mass email to the specified number of email addresses, after the current transaction commits.

Signature
public Void reserveMassEmailCapacity(Integer amountReserved)

Parameters
amountReserved
Type: Integer

Return Value
Type: Void

Usage
This method can be called when you know in advance how many addresses emails will be sent to as a result of the transaction. If the transaction would cause the organization to exceed its daily email limit, using this method results in the following error:
System.HandledException: The daily limit for the org would be exceeded by this request.
If the organization doesn’t have permission to send API or mass email, using this method results in the following error:
System.NoAccessException: The organization is not permitted to send email.

reserveSingleEmailCapacity(amountReserved)
Reserves email capacity to send single email to the specified number of email addresses, after the current transaction commits.

Signature
public Void reserveSingleEmailCapacity(Integer amountReserved)

Parameters
amountReserved
Type: Integer

Return Value
Type: Void

Usage
This method can be called when you know in advance how many addresses emails will be sent to as a result of the transaction. If the transaction would cause the organization to exceed its daily email limit, using this method results in the following error:
System.HandledException: The daily limit for the org would be exceeded by this request.
If the organization doesn’t have permission to send API or mass email, using this method results in the following error:
System.NoAccessException: The organization is not permitted to send email.
the organization doesn’t have permission to send API or mass email, using this method results in the following error:
System.NoAccessException: The organization is not permitted to send email.

**sendEmail(emails, allOrNothing)**

Sends the list of emails instantiated with either SingleEmailMessage or MassEmailMessage and returns a list of SendEmailResult objects. When org preferences are set to save EmailMessage objects and a trigger is defined for EmailMessage objects, the trigger is fired for each SingleEmailMessage individually. The sendEmail method can be called 10 times per Apex transaction and each method invocation can include up to 100 "To", 25 "Cc", and 25 "Bcc" recipients.

**Signature**

```java
public Messaging.SendEmailResult[] sendEmail(Messaging.Email[] emails, Boolean allOrNothing)
```

**Parameters**

- `emails`
  - Type: Messaging.Email[]
- `allOrNothing`
  - Type: Boolean

  The optional `opt_allOrNone` parameter specifies whether sendEmail prevents delivery of all other messages when any of the messages fail due to an error (true), or whether it allows delivery of the messages that don’t have errors (false). The default is true.

**Return Value**

Type: Messaging.SendEmailResult[]

**sendEmailMessage(emailMessageIds, allOrNothing)**

Sends draft email messages as defined by the specified email message IDs and returns a list of SendEmailResult objects.

**Signature**

```java
public Messaging.SendEmailResult[] sendEmailMessage(List<ID> emailMessageIds, Boolean allOrNothing)
```

**Parameters**

- `emailMessageIds`
  - Type: List<ID>
- `allOrNothing`
  - Type: Boolean

**Return Value**

Type: Messaging.SendEmailResult[]

If the `emailMessageIds` parameter is null, the method throws a System.IllegalArgumentException exception.
Usage

The sendEmailMessage method assumes that the optional allOrNothing parameter is always false and ignores the value you set. Delivery of all messages is attempted even if some messages fail due to an error.

The email address of the user calling the sendEmailMessage method is inserted in the From Address field of the email header and the Email Message record.

Example

This example shows how to send a draft email message. It creates a case and a new email message associated with the case. Next, the example sends a draft email message and checks the results. Before running this example, make sure to replace the email address with a valid address.

```apex
Case c = new Case();
insert c;

EmailMessage e = new EmailMessage();
e.parentid = c.id;
// Set to draft status.
// This status is required
// for sendEmailMessage().
e.Status = '5';
e.TextBody = 'Sample email message.';
e.Subject = 'Apex sample';
e.ToAddress = 'customer@email.com';
insert e;
List<Messaging.SendEmailResult> results = Messaging.sendEmailMessage(new ID[]
{ e.id });
System.assertEquals(1, results.size());
System.assertEquals(true,
results[0].success);
```

Versioned Behavior Changes

In API version 54.0 and later, a null emailMessageIds parameter results in a System.IllegalArgumentException exception. In API version 53.0 and earlier, a null emailMessageIds parameter results in an error.

renderEmailTemplate(whoId, whatId, bodies)

Replaces merge fields in text bodies of email templates with values from Salesforce records. Returns an array of RenderEmailTemplateBodyResult objects, each of which corresponds to an element in the supplied array of text bodies. Each RenderEmailTemplateBodyResult provides a success or failure indication, along with either an error code or the rendered text.
Signature

```
public static List<Messaging.RenderEmailTemplateBodyResult> renderEmailTemplate(String whoId, String whatId, List<String> bodies)
```

Parameters

**whoId**
- Type: String
  - The identifier of an object in the database, typically a contact, lead, or user. The database record for that object is read and used in merge field processing.

**whatId**
- Type: String
  - Identifies an object in the database like an account or opportunity. The record for that object is read and used in merge field processing.

**bodies**
- Type: List<String>
  - An array of strings that are examined for merge field references. The corresponding data from the object referenced by the `whoId` or `whatId` replaces the merge field reference.

Return Value

Type: List<Messaging.RenderEmailTemplateBodyResult>

Usage

Use this method in situations in which you want to dynamically compose blocks of text that are enriched with data from the database. You can then use the the rendered blocks of text to compose and send an email or update a text value in another database record.

Executing the `renderEmailTemplate` method counts toward the SOQL governor limit. The number of SOQL queries that this method consumes is the number of elements in the list of strings passed in the `bodies` parameter.

SEE ALSO:
- Execution Governors and Limits

**renderStoredEmailTemplate(templateId, whoId, whatId)**

Renders a text, custom, HTML, or Visualforce email template that exists in the database into an instance of `Messaging.SingleEmailMessage`. Includes all attachment content in the returned email message.

Signature

```
public static Messaging.SingleEmailMessage renderStoredEmailTemplate(String templateId, String whoId, String whatId)
```

Parameters

**templateId**
- Type: String
  - An email template that exists in the database, such as text, HTML, custom, and Visualforce templates.
whoId
Type: String
The identifier of an object in the database, typically a contact, lead, or user. The database record for that object is read and used in merge field processing.

whatId
Type: String
Identifies an object in the database, like an account or opportunity. The record for that object is read and used in merge field processing.

Return Value
Type: Messaging.SingleEmailMessage

Usage
Executing the renderStoredEmailTemplate method counts toward the SOQL governor limit as one query.

SEE ALSO:
Execution Governors and Limits

renderStoredEmailTemplate(templateId, whoId, whatId, attachmentRetrievalOption)
Renders a text, custom, HTML, or Visualforce email template that exists in the database into an instance of Messaging.SingleEmailMessage. Provides options for including attachment metadata only, attachment metadata and content, or excluding attachments.

Signature
public static Messaging.SingleEmailMessage renderStoredEmailTemplate(String templateId, String whoId, String whatId, Messaging.AttachmentRetrievalOption attachmentRetrievalOption)

Parameters

templateId
Type: String
An email template that exists in the database, such as text, HTML, custom, and Visualforce templates.

whoId
Type: String
The identifier of an object in the database, typically a contact, lead, or user. The database record for that object is read and used in merge field processing.

whatId
Type: String
Identifies an object in the database, like an account or opportunity. The record for that object is read and used in merge field processing.
attachmentRetrievalOption
Type: Messaging.AttachmentRetrievalOption

Specifies options for including attachments in the fileAttachments property of the returned Messaging.SingleEmailMessage. Set to one of the Messaging.AttachmentRetrievalOption values to include attachment metadata only, attachment metadata and content, or to exclude attachments.

Note: When the attachmentRetrievalOption parameter is not set to NONE, the entityAttachments property of Messaging.SingleEmailMessage contains the ID of the Salesforce content objects to attach (ContentVersion or Document). The fileAttachments property contains the IDs of attachments, in addition to all the IDs in the entityAttachments property. As a result, the ID values in entityAttachments are duplicates of the IDs in the fileAttachments property. If you call renderStoredEmailTemplate() by passing the METADATA_WITH_BODY option, and send the rendered email message, the email will contain duplicate attachments. Before using the returned email message with sendEmail(emails, allOrNothing), you can remove attachments from fileAttachments that are duplicated in entityAttachments.

Return Value
Type: Messaging.SingleEmailMessage

Usage
Executing the renderStoredEmailTemplate method counts toward the SOQL governor limit as one query.

renderStoredEmailTemplate(templateId, whoId, whatId, attachmentRetrievalOption, updateEmailTemplateUsage)

Renders a text, custom, HTML, or Visualforce email template that exists in the database into an instance of Messaging.SingleEmailMessage. Provides options for including attachment metadata only, attachment metadata and content, or excluding attachments.

Signature
public static Messaging.SingleEmailMessage renderStoredEmailTemplate(String templateId, String whoId, String whatId, Messaging.AttachmentRetrievalOption attachmentRetrievalOption, Boolean updateEmailTemplateUsage)

Parameters

templateId
Type: String

An email template that exists in the database, such as text, HTML, custom, and Visualforce templates.

whoId
Type: String

The identifier of an object in the database, typically a contact, lead, or user. The database record for that object is read and used in merge field processing.

whatId
Type: String

3255
Identifies an object in the database, like an account or opportunity. The record for that object is read and used in merge field processing.

attachmentRetrievalOption
Type: Messaging.AttachmentRetrievalOption

Specifies options for including attachments in the fileAttachments property of the returned Messaging.SingleEmailMessage. Set to one of the Messaging.AttachmentRetrievalOption values to include attachment metadata only, attachment metadata and content, or to exclude attachments.

Note: When the attachmentRetrievalOption parameter is not set to NONE, the entityAttachments property of Messaging.SingleEmailMessage contains the ID of the Salesforce content objects to attach (ContentVersion or Document). The fileAttachments property contains the IDs of attachments, in addition to all the IDs in the entityAttachments property. As a result, the ID values in entityAttachments are duplicates of the IDs in the fileAttachments property. If you call renderStoredEmailTemplate() by passing the METADATA_WITH_BODY option, and send the rendered email message, the email will contain duplicate attachments. Before using the returned email message with sendEmail(emails, allOrNothing), you can remove attachments from fileAttachments that are duplicated in entityAttachments.

updateEmailTemplateUsage
Type: Boolean

Specifies whether the usage field in the EmailTemplate record is updated upon successful rendering.

Return Value
Type: Messaging.SingleEmailMessage

Usage
Executing the renderStoredEmailTemplate method counts toward the SOQL governor limit as one query.

MultiStaticResourceCalloutMock Class
Utility class used to specify a fake response using multiple resources for testing HTTP callouts.

Namespace
System

Usage
Use the methods in this class to set the response properties for testing HTTP callouts. You can specify a resource for each endpoint.

IN THIS SECTION:
MultiStaticResourceCalloutMock Constructors
MultiStaticResourceCalloutMock Methods

MultiStaticResourceCalloutMock Constructors
The following are constructors for MultiStaticResourceCalloutMock.
MultiStaticResourceCalloutMock() creates a new instance of the System.MultiStaticResourceCalloutMock class.

**Signature**

```java
public MultiStaticResourceCalloutMock()
```

**MultiStaticResourceCalloutMock Methods**

The following are methods for MultiStaticResourceCalloutMock. All are instance methods.

**setHeader(headerName, headerValue)**

Sets the specified header name and value for the fake response.

**setStaticResource(endpoint, resourceName)**

Sets the specified static resource corresponding to the endpoint. The static resource contains the response body.

**setStatus(httpStatus)**

Sets the specified HTTP status for the response.

**setStatusCode(httpStatusCode)**

Sets the specified HTTP status code for the response.

**setHeader(headerName, headerValue)**

Sets the specified header name and value for the fake response.

**Signature**

```java
public Void setHeader(String headerName, String headerValue)
```

**Parameters**

- **headerName**
  - Type: String
- **headerValue**
  - Type: String

**Return Value**

- Type: Void
**setStaticResource(endpoint, resourceName)**

Sets the specified static resource corresponding to the endpoint. The static resource contains the response body.

**Signature**

```java
public Void setStaticResource(String endpoint, String resourceName)
```

**Parameters**

- `endpoint`  
  Type: `String`
- `resourceName`  
  Type: `String`

**Return Value**

Type: Void

**setStatus(httpStatus)**

Sets the specified HTTP status for the response.

**Signature**

```java
public Void setStatus(String httpStatus)
```

**Parameters**

- `httpStatus`  
  Type: `String`

**Return Value**

Type: Void

**setStatusCode(httpStatusCode)**

Sets the specified HTTP status code for the response.

**Signature**

```java
public Void setStatusCode(Integer httpStatusCode)
```

**Parameters**

- `httpStatusCode`  
  Type: `Integer`
Return Value
Type: Void

Network Class
Manage Experience Cloud sites.

Namespace
System

IN THIS SECTION:
Network Constructors
Create an instance of the System.Network class.

Network Methods
Get the default landing page, login page, and self-registration page of a site. Asynchronously create site users and records. Get the login and logout URLs for a site. Get a user’s current site. Map dashboards and Insights reports.

Network Constructors
Create an instance of the System.Network class.
The following are constructors for Network.

IN THIS SECTION:
Network()
Creates a new instance of the System.Network class.

public Network()

Network Methods
Get the default landing page, login page, and self-registration page of a site. Asynchronously create site users and records. Get the login and logout URLs for a site. Get a user’s current site. Map dashboards and Insights reports.
The following are methods for Network. All methods are static.

IN THIS SECTION:
communitiesLanding()
Returns a Page Reference to the default landing page for the Experience Cloud site. This is the first tab of the site.
createExternalUserAsync(user, contact, account)
Asynchronously creates an Experience Cloud site user for the given account or contact and associates it with the site. This method processes requests in batches and then sends an email with login information to the user.

createRecordAsync(processType, mbObject)
Asynchronously creates case, lead, and custom object records. This method collects record creation requests and processes them in batches.

forwardToAuthPage(startURL)
Returns a Page Reference to the default login page. StartURL is included as a query parameter for where to redirect after a successful login.

getLoginUrl(networkId)
Returns the absolute URL of the login page used by the Experience Cloud site.

getLogoutUrl(networkId)
Returns the absolute URL of the logout page used by the Experience Cloud site.

getNetworkId()
Returns the user’s current Experience Cloud site.

getSelfRegUrl(networkId)
Returns the absolute URL of the self-registration page used by the Experience Cloud site.

loadAllPackageDefaultNetworkDashboardSettings()
Maps the dashboards from the Salesforce Communities Management package onto each Experience Cloud site’s unconfigured dashboard settings. Returns the number of settings it configures.

loadAllPackageDefaultNetworkPulseSettings()
Maps the Insights reports from the Salesforce Communities Management package onto each Experience Cloud site’s unconfigured Insights settings. Returns the number of settings it configures.

communitiesLanding()
Returns a Page Reference to the default landing page for the Experience Cloud site. This is the first tab of the site.

Signature
public static String communitiesLanding()

Return Value
Type: PageReference

Usage
If digital experiences isn’t enabled for the user’s org or the user is currently in the internal org, returns null.

createExternalUserAsync(user, contact, account)
Asynchronously creates an Experience Cloud site user for the given account or contact and associates it with the site. This method processes requests in batches and then sends an email with login information to the user.
Signature

```java
public static String createExternalUserAsync(SObject user, SObject contact, SObject account)
```

Parameters

**user**
Type: `SObject` (optional)
Information required to create a user.

**contact**
Type: `SObject` (optional)
The contact you want to associate the user with.

**account**
Type: `SObject`
The account you want to associate the user with.

Return Value

Type: `String`
Returns the UUID for the site user.

---

**createRecordAsync**

Asynchronously creates case, lead, and custom object records. This method collects record creation requests and processes them in batches.

Signature

```java
public static String createRecordAsync(String processType, SObject mbObject)
```

Parameters

**processType**
Type: `String`
The process you use to create records.

**mbObject**
Type: `SObject`
The records created for objects. Objects must be supported by the high-volume record creation.

Return Value

Type: `String`
Returns the UUID for the record created.
**forwardToAuthPage(startURL)**

Returns a Page Reference to the default login page. StartURL is included as a query parameter for where to redirect after a successful login.

**Signature**

```java
public static PageReference forwardToAuthPage(String startURL)
```

**Parameters**

- `startURL`
  - Type: `String`

**Return Value**

- Type: `PageReference`

**Usage**

If digital experiences isn’t enabled for the user’s org or the user is currently in the internal org, returns `null`.

**getLoginUrl(networkId)**

Returns the absolute URL of the login page used by the Experience Cloud site.

**Signature**

```java
public static String getLoginUrl(String networkId)
```

**Parameters**

- `networkId`
  - Type: `String`
    - The ID of the Experience Cloud site you’re retrieving this information for.

**Return Value**

- Type: `String`

**Usage**

Returns the full URL for the Lightning Platform or Experience Builder page used as the login page in the Experience Cloud site.

**getLogoutUrl(networkId)**

Returns the absolute URL of the logout page used by the Experience Cloud site.

**Signature**

```java
public static String getLogoutUrl(String networkId)
```
Parameters

networkId
Type: String

The ID of the Experience Cloud site you’re retrieving this information for.

Return Value
Type: String

Usage

Returns the full URL for the Lightning Platform page, Experience Builder page, or Web page used as the logout page in the Experience Cloud site.

getNetworkId()

Returns the user’s current Experience Cloud site.

Signature

public static String getNetworkId()

Return Value
Type: String

Usage

If digital experiences isn’t enabled for the user’s org or the user is currently in the internal org, returns null.

getSelfRegUrl(networkId)

Returns the absolute URL of the self-registration page used by the Experience Cloud site.

Signature

public static String getSelfRegUrl(String networkId)

Parameters

networkId
Type: String

The ID of the Experience Cloud site you’re retrieving this information for.

Return Value
Type: String
Usage

Returns the full URL for the Lightning Platform or Experience Builder page used as the self-registration page in the Experience Cloud site.

`loadAllPackageDefaultNetworkDashboardSettings()`

Maps the dashboards from the Salesforce Communities Management package onto each Experience Cloud site's unconfigured dashboard settings. Returns the number of settings it configures.

Signature

`public static Integer loadAllPackageDefaultNetworkDashboardSettings()`

Return Value

Type: Integer

Usage

If digital experiences is enabled, and the Salesforce Communities Management package is installed, maps the dashboards provided in the package onto each Experience Cloud site's unconfigured dashboard settings. Returns the number of settings it configures. This method is invoked automatically during site creation and package installation, but isn't typically invoked manually.

If digital experiences isn't enabled for the user's org or the user is in the internal org, returns 0.

`loadAllPackageDefaultNetworkPulseSettings()`

Maps the Insights reports from the Salesforce Communities Management package onto each Experience Cloud site's unconfigured Insights settings. Returns the number of settings it configures.

Signature

`public static Integer loadAllPackageDefaultNetworkPulseSettings()`

Return Value

Type: Integer

Usage

If digital experiences is enabled, and the Salesforce Communities Management package is installed, maps the Insights reports provided in the package onto each Experience Cloud site's unconfigured Insights settings. Returns the number of settings it configures. This method is invoked automatically during site creation and package installation, but isn't typically invoked manually.

If digital experiences isn't enabled for the user's org or the user is in the internal org, returns 0.

OrgLimit Class

Contains methods that provide the name, maximum value, and current value of an org limit.
Namespace

System

Usage

Use the System.OrgLimits getAll and getMap methods to obtain either a list or a map of all your org limits. To get details on each limit, use instance methods from System.OrgLimit.

For comparison, the Limits Class returns Apex governor limits and not Salesforce API limits.

Note: Limit values are updated asynchronously, in near-real-time.

IN THIS SECTION:

OrgLimit Methods

OrgLimit Methods

The following are methods for OrgLimit.

IN THIS SECTION:

getLimit()

Returns the maximum allowed limit value.

getName()

Returns the limit’s name.

getValue()

Returns the limit usage value.

toString()

Returns the string representation of the org limit.

getLimit() Returns the maximum allowed limit value.

Signature

public Integer getLimit()

Return Value

Type: Integer

Example

List<System.OrgLimit> limits = OrgLimits.getAll();
for (System.OrgLimit aLimit: limits) {
    System.debug('Limit: ' + aLimit.getName());
System.debug('Max Limit is: ' + aLimit.getLimit());
}

getName()

Returns the limit’s name.

Signature
public String getName()

Return Value
Type: String

Example
List<System.OrgLimit> limits = OrgLimits.getAll();
for (System.OrgLimit aLimit: limits) {
    System.debug('Limit: ' + aLimit.getName());
    System.debug('Max Limit is: ' + aLimit.getLimit());
}

getValue()

Returns the limit usage value.

Signature
public Integer getValue()

Return Value
Type: Integer

Example
List<System.OrgLimit> limits = OrgLimits.getAll();
for (System.OrgLimit aLimit: limits) {
    System.debug('Limit: ' + aLimit.getName());
    System.debug('Usage Value is: ' + aLimit.getValue());
}

toString()

Returns the string representation of the org limit.

Signature
public String toString()
Return Value
Type: String
String denoting the name, current consumption, and maximum value of the org limit. For example:

```
OrgLimit[DailyBulkApiBatches: consumed 25 of 15000]
```

OrgLimits Class
Contains methods that provide a list or map of all OrgLimit instances for Salesforce your org, such as SOAP API requests, Bulk API requests, and Streaming API limits.

Namespace
System

Usage
Use the `System.OrgLimits.getAll` and `getMap` methods to obtain either a list or a map of all your org limits. To get details on each limit, use instance methods from `System.OrgLimit`.
For comparison, the `Limits Class` returns Apex governor limits and not Salesforce API limits.

Note: Limit values are updated asynchronously, in near-real-time.

IN THIS SECTION:
OrgLimits Methods

SEE ALSO:
`REST API Developer Guide: Limits`

OrgLimits Methods
The following are methods for `OrgLimits`.

IN THIS SECTION:
`getAll`
`getMap`

`getAll()`
Returns a list of OrgLimit instances.

`getMap()`
Returns a map of all OrgLimit instances with the limit name as key.

`getAll()`
Returns a list of OrgLimit instances.
Signature

```java
public static List<System.OrgLimit> getAll()
```

Return Value

Type: `List<System.OrgLimit>`

```java
getMap()
```

Returns a map of all OrgLimit instances with the limit name as key.

Signature

```java
public static Map<String, System.OrgLimit> getMap()
```

Return Value

Type: `Map<String, System.OrgLimit>`

Example

```java
Map<String, System.OrgLimit> limitsMap = OrgLimits.getMap();
System.OrgLimit apiRequestsLimit = limitsMap.get('DailyApiRequests');
System.debug('Limit Name: ' + apiRequestsLimit.getName());
System.debug('Usage Value: ' + apiRequestsLimit.getValue());
System.debug('Maximum Limit: ' + apiRequestsLimit.getLimit());
```

**PageReference Class**

A PageReference is a reference to an instantiation of a page. Among other attributes, PageReferences consist of a URL and a set of query parameter names and values.

**Namespace**

`System`

Use a PageReference object:

- To view or set query string parameters and values for a page
- To navigate the user to a different page as the result of an action method

**Instantiation**

In a custom controller or controller extension, you can refer to or instantiate a PageReference in one of the following ways:
PageReference Class

- **Page.**existingPageName

Refers to a PageReference for a Visualforce page that has already been saved in your organization. By referring to a page in this way, the platform recognizes that this controller or controller extension is dependent on the existence of the specified page and will prevent the page from being deleted while the controller or extension exists.

```object
PageReference pageRef = new PageReference('partialURL');
```

Creates a PageReference to any page that is hosted on the Lightning platform. For example, setting 'partialURL' to '/apex/HelloWorld' refers to the Visualforce page located at http://mySalesforceInstance/apex/HelloWorld. Likewise, setting 'partialURL' to '/' + 'recordID' refers to the detail page for the specified record.

This syntax is less preferable for referencing other Visualforce pages than **Page.**existingPageName because the PageReference is constructed at runtime, rather than referenced at compile time. Runtime references are not available to the referential integrity system. Consequently, the platform doesn’t recognize that this controller or controller extension is dependent on the existence of the specified page and won’t issue an error message to prevent user deletion of the page.

```object
PageReference pageRef = new PageReference('fullURL');
```

Creates a PageReference for an external URL. For example:

```object
PageReference pageRef = new PageReference('http://www.google.com');
```

You can also instantiate a PageReference object for the current page with the currentPage ApexPages method. For example:

```object
PageReference pageRef = ApexPages.currentPage();
```

### Request Headers

The following table is a non-exhaustive list of headers that are set on requests.

<table>
<thead>
<tr>
<th>Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host</td>
<td>The host name requested in the request URL. This header is always set on Lightning Platform Site requests and My Domain requests. This header is optional on other requests when HTTP/1.0 is used instead of HTTP/1.1.</td>
</tr>
<tr>
<td>Referer</td>
<td>The URL that is either included or linked to the current request’s URL. This header is optional.</td>
</tr>
<tr>
<td>User-Agent</td>
<td>The name, version, and extension support of the program that initiated this request, such as a web browser. This header is optional and can be overridden in most browsers to be a different value. Therefore, this header should not be relied upon.</td>
</tr>
<tr>
<td>CipherSuite</td>
<td>If this header exists and has a non-blank value, this means that the request is using HTTPS. Otherwise, the request is using HTTP. The contents of a non-blank value are not defined by this API, and can be changed without notice.</td>
</tr>
<tr>
<td>X-Salesforce-SIP</td>
<td>The source IP address of the request. This header is always set on HTTP and HTTPS requests that are initiated outside of Salesforce’s data centers.</td>
</tr>
</tbody>
</table>

**Note:** If a request passes through a content delivery network (CDN) or proxy server, the source IP address might be altered, and no longer the original client IP address.
Example: Retrieving Query String Parameters

The following example shows how to use a PageReference object to retrieve a query string parameter in the current page URL. In this example, the `getAccount` method references the `id` query string parameter:

```java
public with sharing class MyController {
    public Account getAccount() {
        return [SELECT Id, Name FROM Account WITH SECURITY_ENFORCED
            WHERE Id = :ApexPages.currentPage().getParameters().get('Id')];
    }
}
```

The following page markup calls the `getAccount` method from the controller above:

```xml
<apex:page controller="MyController">
    <apex:pageBlock title="Retrieving Query String Parameters">
        You are viewing the {!account.name} account.
    </apex:pageBlock>
</apex:page>
```

Note: For this example to render properly, you must associate the Visualforce page with a valid account record in the URL. For example, if 001D000000IRt53 is the account ID, the resulting URL should be:

```
https://Salesforce_instance/apex/MyFirstPage?id=001D000000IRt53
```

The `getAccount` method uses an embedded SOQL query to return the account specified by the `id` parameter in the URL of the page. To access `id`, the `getAccount` method uses the `ApexPages` namespace:

- First the `currentPage` method returns the `PageReference` instance for the current page. `PageReference` returns a reference to a Visualforce page, including its query string parameters.
- Using the page reference, use the `getParameters` method to return a map of the specified query string parameter names and values.
- Then a call to the `get` method specifying `id` returns the value of the `id` parameter itself.

Example: Navigating to a New Page as the Result of an Action Method

Any action method in a custom controller or controller extension can return a PageReference object as the result of the method. If the `redirect` attribute on the PageReference is set to `true`, the user navigates to the URL specified by the PageReference.

The following example shows how this can be implemented with a `save` method. In this example, the PageReference returned by the `save` method redirects the user to the detail page for the account record that was just saved:

```java
public class mySecondController {
    Account account;

    public Account getAccount() {
        if(account == null) account = new Account();
        return account;
    }
}
```
public PageReference save() {
    // Add the account to the database.
    insert account;
    // Send the user to the detail page for the new account.
    acctPage.setRedirect(true);
    return acctPage;
}

The following page markup calls the `save` method from the controller above. When a user clicks `Save`, he or she is redirected to the detail page for the account just created:

```apex
<apex:page controller="mySecondController" tabStyle="Account">
    <apex:sectionHeader title="New Account Edit Page" />
    <apex:form>
        <apex:pageBlock title="Create a New Account">
            <apex:pageBlockButtons location="bottom">
                <apex:commandButton action="{!save}" value="Save"/>
            </apex:pageBlockButtons>
            <apex:pageBlockSection title="Account Information">
                <apex:inputField id="accountName" value="{!account.name}"/>
                <apex:inputField id="accountSite" value="{!account.site}"/>
            </apex:pageBlockSection>
        </apex:pageBlock>
    </apex:form>
</apex:page>
```

**Example: Redirect Users to a Replacement Experience Cloud Site**

The following example shows how to redirect a user attempting to access a retired feedback site to a self-service help site. If the `redirect` attribute is set to `true` on the PageReference for the feedback site, the user navigates to the URL specified by the PageReference. The `redirectCode` attribute defines the redirection type for search engine optimization in public Experience Cloud sites.

```apex
public class RedirectController {
    // Redirect users to the self-service help site
    public PageReference redirect() {
        final PageReference target = new PageReference(Site.getBaseSecureUrl() + '/SiteLogin');
        target.setRedirect(true);
        // This is a permanent redirection
        target.setRedirectCode(301);
        return target;
    }
}
```

The following example shows how to call the RedirectController class from the retired site page.

```apex
<apex:page controller="RedirectController" action="{!redirect}"/>
```
IN THIS SECTION:

PageReference Constructors
PageReference Methods

PageReference Constructors
The following are constructors for PageReference.

IN THIS SECTION:

PageReference(partialURL)
Creates a new instance of the PageReference class using the specified URL.

PageReference(record)
Generate a new instance of the PageReference class for the specified sObject record.

PageReference (partialURL)
Creates a new instance of the PageReference class using the specified URL.

Signature
public PageReference(String partialURL)

Parameters
partialURL
Type: String

The partial URL of a page hosted on the Lightning Platform or a full external URL. The following are some examples of the partialURL parameter values:
- /recordID: refers to the detail page of a specified record.
- http://www.google.com: refers to an external URL.

PageReference (record)
Generate a new instance of the PageReference class for the specified sObject record.

Signature
public PageReference(SObject record)

Parameters
record
Type: SObject
The sObject record that references the ApexPage. The reference must be an ApexPage.

SEE ALSO:

Visualforce Developer Guide: apexpage
SOAP API Developer Guide: ApexPage

PageReference Methods
The following are methods for PageReference. All are instance methods.

IN THIS SECTION:

forResource(resourceName, path)
Create a PageReference for nested content inside a zip static resource, by name and path.

forResource(resourceName)
Create a PageReference for a static resource, by name.

getAnchor()
Returns the name of the anchor referenced in the page’s URL. That is, the part of the URL after the hashtag (#).

getContent()
Returns the output of the page, as displayed to a user in a web browser.

getContentAsPDF()
Returns the page in PDF, regardless of the <apex:page> component’s renderAs attribute.

getCookies()
Returns a map of cookie names and cookie objects, where the key is a String of the cookie name and the the value contains the cookie object with that name.

getHeaders()
Returns a map of the request headers, where the key string contains the name of the header, and the value string contains the value of the header.

getParameters()
Returns a map of the query string parameters for the PageReference; both POST and GET parameters are included. The key string contains the name of the parameter, while the value string contains the value of the parameter.

getRedirect()
Returns the current value of the PageReference object’s redirect attribute.

getRedirectCode()
Returns the HTTP redirect code used when getRedirect() is set to true for the PageReference object.

getUrl()
Returns the relative URL associated with the PageReference when it was originally defined, including any query string parameters and anchors.

setAnchor(ANCHOR)
Sets the URL’s anchor reference to the specified string.

setCookies(cookies)
Creates a list of cookie objects. Used in conjunction with the Cookie class.
setRedirect(redirect)
Sets the value of the PageReference object's `redirect` attribute. If set to `true`, a redirect is performed through a client side redirect.

setRedirectCode(redirectCode)
Sets the HTTP redirect code to use for the PageReference object when `setRedirect(redirect)` is set to `true`.

**forResource(resourceName, path)**
Create a PageReference for nested content inside a zip static resource, by name and path.

**Signature**

```java
public static System.PageReference forResource(String resourceName, String path)
```

**Parameters**

- `resourceName`
  - Type: `String`
  - The resource name

- `path`
  - Type: `String`
  - The resource path

**Return Value**

Type: `System.PageReference`

**forResource(resourceName)**
Create a PageReference for a static resource, by name.

**Signature**

```java
public static System.PageReference forResource(String resourceName)
```

**Parameters**

- `resourceName`
  - Type: `String`
  - The resource name

**Return Value**

Type: `System.PageReference`

**getAnchor()**
Returns the name of the anchor referenced in the page's URL. That is, the part of the URL after the hashtag (`#`).
Signature

```java
public String getAnchor()
```

Return Value

Type: String

⚠️ Note: Instances of PageReference returned by ApexPages.currentPage() have a null anchor attribute, because URL fragments are not sent to the Salesforce server during a request.

**getContent()**

Returns the output of the page, as displayed to a user in a web browser.

Signature

```java
public Blob getContent()
```

Return Value

Type: Blob

**Usage**

The content of the returned Blob depends on how the page is rendered. If the page is rendered as a PDF file, it returns the PDF document. If the page is not rendered as PDF, it returns HTML. To access the content of the returned HTML as a string, use the `toString Blob` method.

⚠️ Note: If you use `getContent` in a test method, the test method fails. `getContent` is treated as a callout in API version 34.0 and later.

This method can’t be used in:

- Triggers
- Test methods
- Apex email services

If the Visualforce page has an error, an `ExecutionException` is thrown.

**getContentAsPDF()**

Returns the page in PDF, regardless of the `<apex:page>` component’s `renderAs` attribute.

Signature

```java
public Blob getContentAsPDF()
```

Return Value

Type: Blob
Usage

Note: If you use `getContentAsPDF` in a test method, the test method fails. `getContentAsPDF` is treated as a callout in API version 34.0 and later.

This method can’t be used in:
- Triggers
- Test methods
- Apex email services

**getCookies()**

Returns a map of cookie names and cookie objects, where the key is a String of the cookie name and the the value contains the cookie object with that name.

**Signature**

```java
public Map<String, System.Cookie> getCookies()
```

**Return Value**

Type: `Map<String, System.Cookie>`

**Usage**

Used in conjunction with the `Cookie` class. Only returns cookies with the “apex__” prefix set by the `setCookies` method.

**getHeaders()**

Returns a map of the request headers, where the key string contains the name of the header, and the value string contains the value of the header.

**Signature**

```java
public Map<String, String> getHeaders()
```

**Return Value**

Type: `Map<String, String>`

**Usage**

This map can be modified and remains in scope for the PageReference object. For instance, you could do:

```java
PageReference.getHeaders().put('Date', '9/9/99');
```

For a description of request headers, see Request Headers.

**getParameters()**

Returns a map of the query string parameters for the PageReference; both POST and GET parameters are included. The key string contains the name of the parameter, while the value string contains the value of the parameter.
Signature

public Map<String, String> getParameters()

Return Value

Type: Map<String, String>

Usage

This map can be modified and remains in scope for the PageReference object. For instance, you could do:

```java
PageReference.getParameters().put('id', myID);
```

Parameter keys are case-insensitive. For example:

```java
System.assert(
    ApexPages.currentPage().getParameters().get('myParamName') ==
    ApexPages.currentPage().getParameters().get('myparamname'));
```

getRedirect()

Returns the current value of the PageReference object's redirect attribute.

Signature

public Boolean getRedirect()

Return Value

Type: Boolean

Usage

Note that if the URL of the PageReference object is set to a website outside of the salesforce.com domain, the redirect always occurs, regardless of whether the redirect attribute is set to true or false.

getRedirectCode()

Returns the HTTP redirect code used when getRedirect() is set to true for the PageReference object.

Signature

public Integer getRedirectCode()

Return Value

Type: Integer

Possible Values:

- 0 — Redirect using the default redirect action for this PageReference. Typically a JavaScript-based redirection or HTTP 302.
Note: Site URL Rewriter Interface implementations pointing to a PageReference with a redirectCode of 0 are not redirected.

- **301** — Moved Permanently. Redirect users by sending an HTTP GET request to the target location. Includes instructions to update any references to the requested URL with the target location.
- **302** — Moved Temporarily. Redirect users by sending an HTTP GET request to the target location. Because the redirection is temporary, it doesn’t include update instructions.
- **303** — See Other. Redirect users by sending an HTTP GET request to the target location. Not commonly used. Useful when the client sends a POST request and you want the client to call the new web page using a GET request instead of a POST request.
- **307** — Temporary Redirect. Send the same HTTP request, regardless of the HTTP method, to the target location. Because the redirection is temporary, it doesn’t include update instructions.
- **308** — Permanent Redirect. Send the same HTTP request, regardless of the HTTP method, to the target location. Includes instructions to update any references to the requested URL with the target location.

**getUrl()**

Returns the relative URL associated with the PageReference when it was originally defined, including any query string parameters and anchors.

**Signature**

```java
public String getUrl()
```

**Return Value**

Type: `String`

**setAnchor(anchor)**

Sets the URL’s anchor reference to the specified string.

**Signature**

```java
public System.PageReference setAnchor(String anchor)
```

**Parameters**

- `anchor`
  
  Type: `String`

**Return Value**

Type: `System.PageReference`

**setCookies(cookies)**

Creates a list of cookie objects. Used in conjunction with the `Cookie` class.
**setCookies(cookies)**

*Signature*

```java
public Void setCookies(Cookie[] cookies)
```

*Parameters*

- `cookies`  
  Type: `System.Cookie[]`

*Return Value*

Type: Void

*Usage*

**Important:**

- Cookie names and values set in Apex are URL encoded, that is, characters such as @ are replaced with a percent sign and their hexadecimal representation.
- The `setCookies` method adds the prefix "apex__" to the cookie names.
- Setting a cookie's value to `null` sends a cookie with an empty string value instead of setting an expired attribute.
- After you create a cookie, the properties of the cookie can't be changed.
- Be careful when storing sensitive information in cookies. Pages are cached regardless of a cookie value. If you use a cookie value to generate dynamic content, you should disable page caching. For more information, see Configure Site Caching in Salesforce Help.

**setRedirect(redirect)**

Sets the value of the `PageReference` object's `redirect` attribute. If set to `true`, a redirect is performed through a client side redirect.

*Signature*

```java
public System.PageReference setRedirect(Boolean redirect)
```

*Parameters*

- `redirect`  
  Type: `Boolean`

*Return Value*

Type: `System.PageReference`

*Usage*

This type of redirect performs an HTTP GET request, and flushes the view state, which uses POST. If set to `false`, the redirect is a server-side forward that preserves the view state if and only if the target page uses the same controller and contains the proper subset of extensions used by the source page.
Note that if the URL of the PageReference object is set to a website outside of the salesforce.com domain, or to a page with a different controller or controller extension, the redirect always occurs, regardless of whether the redirect attribute is set to true or false.

```java
setRedirectCode(redirectCode)
```

Sets the HTTP redirect code to use for the PageReference object when setRedirect(redirect) is set to true.

**Signature**

```java
```

**Parameters**

- **redirectCode**
  - Type: Integer
  - Valid values:
    - 0 — Redirect using the default redirect action for this PageReference. Typically a JavaScript-based redirection or HTTP 302.
      
      ※ **Note:** Site URLRewriter Interface implementations pointing to a PageReference with a redirectCode of 0 are not redirected.
    
    - 301 — Moved Permanently. Redirect users by sending an HTTP GET request to the target location. Includes instructions to update any references to the requested URL with the target location.
    
    - 302 — Moved Temporarily. Redirect users by sending an HTTP GET request to the target location. Because the redirection is temporary, it doesn't include update instructions.
    
    - 303 — See Other. Redirect users by sending an HTTP GET request to the target location. Not commonly used. Useful when the client sends a POST request and you want the client to call the new web page using a GET request instead of a POST request.
    
    - 307 — Temporary Redirect. Send the same HTTP request, regardless of the HTTP method, to the target location. Because the redirection is temporary, it doesn't include update instructions.
    
    - 308 — Permanent Redirect. Send the same HTTP request, regardless of the HTTP method, to the target location. Includes instructions to update any references to the requested URL with the target location.
      
      If the redirect code contains an invalid integer, an error message is displayed when PageReference is used by Salesforce for redirection.

**Return Value**

- Type: System.PageReference

### Packaging Class

Contains a method for obtaining information about managed and unlocked packages.

**Namespace**

System
Usage
In the context of a package, use the getCurrentPackageId method to retrieve the packageId.

Packaging Methods
The following are methods for Packaging.

getCurrentPackageId()
Returns the context packageId in managed and unlocked packages.

Signature
public String getCurrentPackageId()

Return Value
Type: String

Usage
For managed packages, this method can be combined with isCurrentUserLicensedForPackage(packageId) to retrieve the packageId at runtime. Then, use packageId to confirm that the contextual user is licensed to use that managed package.

Pattern Class
Represents a compiled representation of a regular expression.

Namespace
System

Pattern Methods
The following are methods for Pattern.
matcher(stringToMatch)
Creates a Matcher object that matches the input string *stringToMatch* against this Pattern object.

matches(regExp, stringToMatch)
Compiles the regular expression *regExp* and tries to match it against the specified string. This method returns *true* if the specified string matches the regular expression, *false* otherwise.

pattern()
Returns the regular expression from which this Pattern object was compiled.

quote(yourString)
Returns a string that can be used to create a pattern that matches the string *yourString* as if it were a literal pattern.

split(regExp)
Returns a list that contains each substring of the String that matches this pattern.

split(regExp, limit)
Returns a list that contains each substring of the String that is terminated either by the regular expression *regExp* that matches this pattern, or by the end of the String.

**compile(regExp)**
Compiles the regular expression into a Pattern object.

**Signature**

```java
public static Pattern compile(String regExp)
```

**Parameters**

*regExp*
Type: *String*

**Return Value**

Type: *System.Pattern*

**matcher(stringToMatch)**
Creates a Matcher object that matches the input string *stringToMatch* against this Pattern object.

**Signature**

```java
public Matcher matcher(String stringToMatch)
```

**Parameters**

*stringToMatch*
Type: *String*

**Return Value**

Type: *Matcher*
matches(regExp, stringtoMatch)
Compiles the regular expression `regExp` and tries to match it against the specified string. This method returns `true` if the specified string matches the regular expression, `false` otherwise.

**Signature**
```
public static Boolean matches(String regExp, String stringtoMatch)
```

**Parameters**
- `regExp`
  Type: `String`
- `stringtoMatch`
  Type: `String`

**Return Value**
Type: `Boolean`

**Usage**
If a pattern is to be used multiple times, compiling it once and reusing it is more efficient than invoking this method each time.

**Example**
Note that the following code example:
```
Pattern.matches(regExp, input);
```
produces the same result as this code example:
```
Pattern.compile(regex).matcher(input).matches();
```

pattern()
Returns the regular expression from which this Pattern object was compiled.

**Signature**
```
public String pattern()
```

**Return Value**
Type: `String`

quote(yourString)
Returns a string that can be used to create a pattern that matches the string `yourString` as if it were a literal pattern.
public static String quote(String yourString)

Parameters

yourString
  Type: String

Return Value

Type: String

Usage

Metacharacters (such as $ or ^) and escape sequences in the input string are treated as literal characters with no special meaning.

split(regExp)

Returns a list that contains each substring of the String that matches this pattern.

Signature

public String[] split(String regExp)

Parameters

regExp
  Type: String

Return Value

Type: String[]

Note: In API version 34.0 and earlier, a zero-width regExp value produces an empty list item at the beginning of the method’s output.

Usage

The substrings are placed in the list in the order in which they occur in the String. If regExp does not match the pattern, the resulting list has just one element containing the original String.

split(regExp, limit)

Returns a list that contains each substring of the String that is terminated either by the regular expression regExp that matches this pattern, or by the end of the String.

Signature

public String[] split(String regExp, Integer limit)
Parameters

- **regExp**
  - Type: String

- **limit**
  - Type: Integer
  
  (Optional) Controls the number of times the pattern is applied and therefore affects the length of the list.
  
  - If `limit` is greater than zero:
    - The pattern is applied a maximum of `(limit - 1)` times.
    - The list's length is no greater than `limit`.
    - The list's last entry contains all input beyond the last matched delimiter.
  
  - If `limit` is non-positive, the pattern is applied as many times as possible, and the list can have any length.
  
  - If `limit` is zero, the pattern is applied as many times as possible, the list can have any length, and trailing empty strings are discarded.

Return Value

- Type: String[]

**Note:** In API version 34.0 and earlier, a zero-width `regExp` value produces an empty list item at the beginning of the method's output.

Queueable Interface

Enables the asynchronous execution of Apex jobs that can be monitored.

Namespace

- System

Usage

To execute Apex as an asynchronous job, implement the **Queueable** interface and add the processing logic in your implementation of the `execute` method.

To implement the **Queueable** interface, you must first declare a class with the `implements` keyword as follows:

```java
public class MyQueueableClass implements Queueable {

    // Your code here
}
```

Next, your class must provide an implementation for the following method:

```java
public void execute(QueueableContext context) {
    // Your code here
}
```

Your class and method implementation must be declared as `public` or `global`.

To submit your class for asynchronous execution, call the `System.enqueueJob` by passing it an instance of your class implementation of the **Queueable** interface as follows:

```java
ID jobID = System.enqueueJob(new MyQueueableClass());
```
Queueable Methods

The following are methods for Queueable.

IN THIS SECTION:

execute(context)
Executes the queueable job.

**execute (context)**
Executes the queueable job.

**Signature**

```
public void execute(QueueableContext context)
```

**Parameters**

- `context`
  - Type: QueueableContext
  - Contains the job ID.

**Return Value**

Type: Void

Queueable Example Implementation

This example is an implementation of the Queueable interface. The `execute` method in this example inserts a new account.

```
public class AsyncExecutionExample implements Queueable {
    public void execute(QueueableContext context) {
        Account a = new Account(Name='Acme',Phone='(415) 555-1212');
        insert a;
    }
}
```

To add this class as a job on the queue, call this method:

```
ID jobID = System.enqueueJob(new AsyncExecutionExample());
```
After you submit your queueable class for execution, the job is added to the queue and will be processed when system resources become available. You can monitor the status of your job programmatically by querying AsyncApexJob or through the user interface in Setup by entering Apex Jobs in the Quick Find box, then selecting Apex Jobs.

To query information about your submitted job, perform a SOQL query on AsyncApexJob by filtering on the job ID that the System.enqueueJob method returns. This example uses the jobID variable that was obtained in the previous example.

```apex
AsyncApexJob jobInfo = [SELECT Status,NumberOfErrors FROM AsyncApexJob WHERE Id=:jobID];
```

Similar to future jobs, queueable jobs don’t process batches, and so the number of processed batches and the number of total batches are always zero.

### Testing Queueable Jobs

This example shows how to test the execution of a queueable job in a test method. A queueable job is an asynchronous process. To ensure that this process runs within the test method, the job is submitted to the queue between the Test.startTest and Test.stopTest block. The system executes all asynchronous processes started in a test method synchronously after the Test.stopTest statement. Next, the test method verifies the results of the queueable job by querying the account that the job created.

```apex
@isTest
class AsyncExecutionExampleTest {
    static testmethod void test1() {
        // startTest/stopTest block to force async processes
        // to run in the test.
        Test.startTest();
        System.enqueueJob(new AsyncExecutionExample());
        Test.stopTest();

        // Validate that the job has run
        // by verifying that the record was created.
        // This query returns only the account created in test context by the
        // Queueable class method.
        Account acct = [SELECT Name,Phone FROM Account WHERE Name='Acme' LIMIT 1];
        System.assertNotEquals(null, acct);
        System.assertEquals('(415) 555-1212', acct.Phone);
    }
}
```

**Note:** The ID of a queueable Apex job isn’t returned in test context—System.enqueueJob returns null in a running test.

---

### QueueableContext Interface

Represents the parameter type of the `execute()` method in a class that implements the Queueable interface and contains the job ID. This interface is implemented internally by Apex.

**Namespace**

System
QueueableContext Methods

The following are methods for QueueableContext.

IN THIS SECTION:

**getJobId()**
Returns the ID of the submitted job that uses the Queueable interface.

**getJobId()**
Returns the ID of the submitted job that uses the Queueable interface.

**Signature**

```java
public ID getJobId()
```

**Return Value**
Type: ID
The ID of the submitted job.

QueueableDuplicateSignature Class

Used in the AsyncOptions class to store the queueable job signature in the DuplicateSignature property.

**Namespace**

System

**In This Section:**

QueueableDuplicateSignature Methods

QueueableDuplicateSignature Methods

The following are methods for QueueableDuplicateSignature.

IN THIS SECTION:

**toString()**
Returns the duplicate signature as a string value.

**toString()**
Returns the duplicate signature as a string value.
Signature

public String toString()

Return Value

Type: String

QueueableDuplicateSignature.Builder Class

Build a unique signature for your queueable job using this inner builder class. The build() class method builds a QueueableDuplicateSignature object with input from the addId(), addInteger(), and addString() methods. Use the DuplicateSignature property in the AsyncOptions class to store the queueable job signature. Enqueue your job by using the System.enqueueJob() with the AsyncOptions parameter.

Namespace

System

Examples

This example builds the async job signature with UserId and the string MyQueueable.

```java
AsyncOptions options = new AsyncOptions();
options.DuplicateSignature = new System.QueueableDuplicateSignature.Builder()
    .addId(UserInfo.getUserId())
    .addString('MyQueueable')
    .build();
try {
    System.enqueueJob(new MyQueueable(), options);
} catch (DuplicateMessageException ex) {
    //Exception is thrown if there is already an enqueued job with the same signature
    Assert.areEqual('Attempt to enqueue job with duplicate queueable signature', ex.getMessage());
}
```

This example builds the async job signature using ApexClass Id and the hash value of an sObject.

```java
AsyncOptions options = new AsyncOptions();
options.DuplicateSignature = new QueueableDuplicateSignature.Builder()
    .addInteger(System.hashCode(someAccount))
    .addId([SELECT Id FROM ApexClass
            WHERE Name='MyQueueable'].Id)
    .build();
System.enqueueJob(new MyQueueable(), options);
```

IN THIS SECTION:

QueueableDuplicateSignature.Builder Methods
QueueableDuplicateSignature.Builder Methods

The following are methods for QueueableDuplicateSignature.Builder.

IN THIS SECTION:

addId(inputId)
Adds an ID to build a unique signature for a queueable job. You can then enqueue the job by using the signature as the AsyncOptions parameter to System.enqueueJob().

addInteger(inputInteger)
Adds an integer to build a unique signature for a queueable job. You can then enqueue the job by using the signature as the AsyncOptions parameter to System.enqueueJob().

addString(inputString)
Adds a string to build a unique signature for a queueable job. You can then enqueue the job by using the signature as the AsyncOptions parameter to System.enqueueJob().

build()
Builds a unique signature for a queueable job. You can then enqueue the job by using the signature as the AsyncOptions parameter to System.enqueueJob().

getMaxSize()
Gets the maximum size of the queueable job signature in bytes.

getRemainingSize()
Gets the remaining size of the queueable job signature in bytes, after subtracting what is already used by the signature from the maximum allowed number.

getSize()
Gets the size of the queueable job signature in bytes.

addId (inputId)
Adds an ID to build a unique signature for a queueable job. You can then enqueue the job by using the signature as the AsyncOptions parameter to System.enqueueJob().

Signature

public System.QueueableDuplicateSignature.Builder addId(Id id)

Parameters

inputId
Type: Id

Return Value

Type: QueueableDuplicateSignature.Builder
addInteger(inputInteger)

Adds an integer to build a unique signature for a queueable job. You can then enqueue the job by using the signature as the AsyncOptions parameter to System.enqueueJob().

Signature

public System.QueueableDuplicateSignature.Builder addInteger(Integer i)

Parameters

inputInteger
Type: Integer

Return Value
Type: QueueableDuplicateSignature.Builder

addString(inputString)

Adds a string to build a unique signature for a queueable job. You can then enqueue the job by using the signature as the AsyncOptions parameter to System.enqueueJob().

Signature

public System.QueueableDuplicateSignature.Builder addString(String s)

Parameters

inputString
Type: String

Return Value
Type: QueueableDuplicateSignature.Builder

build()

Builds a unique signature for a queueable job. You can then enqueue the job by using the signature as the AsyncOptions parameter to System.enqueueJob().

Signature

public System.QueueableDuplicateSignature build()

Return Value
Type: QueueableDuplicateSignature Class
getMaxSize()
Gets the maximum size of the queueable job signature in bytes.

Signature
public Integer getMaxSize()

Return Value
Type: Integer

getRemainingSize()
Gets the remaining size of the queueable job signature in bytes, after subtracting what is already used by the signature from the maximum allowed number.

Signature
public Integer getRemainingSize()

Return Value
Type: Integer

getSize()
Gets the size of the queueable job signature in bytes.

Signature
public Integer getSize()

Return Value
Type: Integer

QuickAction Class
Use Apex to request and process actions on objects that allow custom fields, on objects that appear in a Chatter feed, or on objects that are available globally.

Namespace
System
**Example**

In this sample, the trigger determines if the new contacts to be inserted are created by a quick action. If so, it sets the `WhereFrom__c` custom field to a value that depends on whether the quick action is global or local to the contact. Otherwise, if the inserted contacts don’t originate from a quick action, the `WhereFrom__c` field is set to 'NoAction'.

```apex
trigger accTrig2 on Contact (before insert) {
    for (Contact c : Trigger.new) {
        if (c.getQuickActionName() == QuickAction.CreateContact) {
            c.WhereFrom__c = 'GlobalAction1';
        } else if (c.getQuickActionName() == Schema.Account.QuickAction.CreateContact) {
            c.WhereFrom__c = 'AccountAction';
        } else if (c.getQuickActionName() == null) {
            c.WhereFrom__c = 'NoAction';
        } else {
            System.assert(false);
        }
    }
}
```

This sample performs a global action—QuickAction.CreateContact—on the passed-in contact object.

```apex
public Id globalCreate(Contact c) {
    QuickAction.QuickActionRequest req = new QuickAction.QuickActionRequest();
    req.quickActionName = QuickAction.CreateContact;
    req.record = c;
    QuickAction.QuickActionResult res = QuickAction.performQuickAction(req);
    return c.id;
}
```

**SEE ALSO:**
- QuickActionRequest Class
- QuickActionResult Class

**QuickAction Methods**

The following are methods for QuickAction. All methods are static.

**IN THIS SECTION:**
- `describeAvailableQuickActions(parentType)`
  Returns metadata information for the available quick actions of the provided parent object.
- `describeAvailableQuickActions(sObjectNames)`
  Returns the metadata information for the provided quick actions.
- `performQuickAction(quickActionRequest)`
  Performs the quick action specified in the quick action request and returns the action result.
- `performQuickAction(quickActionRequest, allOrNothing)`
  Performs the quick action specified in the quick action request with the option for partial success, and returns the result.
- `performQuickActions(quickActionRequests)`
  Performs the quick actions specified in the quick action request list and returns action results.
performQuickActions(quickActionRequests, allOrNothing)
Performs the quick actions specified in the quick action request list with the option for partial success, and returns action results.

describeAvailableQuickActions(\texttt{parentType})
Returns metadata information for the available quick actions of the provided parent object.

\textbf{Signature}
\begin{verbatim}
public static List\<QuickAction.DescribeAvailableQuickActionResult\> describeAvailableQuickActions(\texttt{String parentType})
\end{verbatim}

\textbf{Parameters}
\begin{verbatim}
parentType
Type: String
\end{verbatim}

The parent object type. This can be an object type name (\texttt{Account}) or \texttt{Global} (meaning that this method is called at a global level and not an entity level).

\textbf{Return Value}
Type: \texttt{List<QuickAction.DescribeAvailableQuickActionResult>}
The metadata information for the available quick actions of the parent object.

\textbf{Example}
\begin{verbatim}
// Called for Account entity.
List<QuickAction.DescribeAvailableQuickActionResult> result1 =
QuickAction.DescribeAvailableQuickActions('Account');

// Called at global level, not entity level.
List<QuickAction.DescribeAvailableQuickActionResult> result2 =
QuickAction.DescribeAvailableQuickActions('Global');
\end{verbatim}

describeAvailableQuickActions(\texttt{sObjectNames})
Returns the metadata information for the provided quick actions.

\textbf{Signature}
\begin{verbatim}
public static List\<QuickAction.DescribeQuickActionResult\> describeAvailableQuickActions(List\<String\> sObjectNames)
\end{verbatim}

\textbf{Parameters}
\begin{verbatim}
sObjectNames
Type: List\<String\>
\end{verbatim}

The names of the quick actions. The quick action name can contain the entity name if it is at the entity level ('Account.QuickCreateContact'), or 'Global' if used for the action at the global level ('Global.CreateNewContact').
Return Value
Type: List<QuickAction.DescribeQuickActionResult>
The metadata information for the provided quick actions.

Example

```java
// First 3 parameter values are for actions at the entity level.
// Last parameter is for an action at the global level.
List<QuickAction.DescribeQuickActionResult> result =
    QuickAction.DescribeQuickActions(new List<String> {
        'Account.QuickCreateContact', 'Opportunity.Update1',
        'Contact.Create1', 'Global.CreateNewContact' });
```

`performQuickAction(quickActionRequest)`
Performs the quick action specified in the quick action request and returns the action result.

Signature
```
public static QuickAction.QuickActionResult performQuickAction(QuickAction.QuickActionRequest quickActionRequest)
```

Parameters
`quickActionRequest`
Type: QuickAction.QuickActionRequest

Return Value
Type: QuickAction.QuickActionResult

`performQuickAction(quickActionRequest, allOrNothing)`
Performs the quick action specified in the quick action request with the option for partial success, and returns the result.

Signature
```
public static QuickAction.QuickActionResult performQuickAction(QuickAction.QuickActionRequest quickActionRequest, Boolean allOrNothing)
```

Parameters
`quickActionRequest`
Type: QuickAction.QuickActionRequest
`allOrNothing`
Type: Boolean
Specifies whether this operation allows partial success. If you specify false for this argument and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why.

Return Value
Type: QuickAction.QuickActionResult

**performQuickActions(quickActionRequests)**
Performs the quick actions specified in the quick action request list and returns action results.

**Signature**
```java
public static List<QuickAction.QuickActionResult> performQuickActions(List<QuickAction.QuickActionRequest> quickActionRequests)
```

**Parameters**
- `quickActionRequests`
  Type: List<QuickAction.QuickActionRequest>

**Return Value**
Type: List<QuickAction.QuickActionResult>

**performQuickActions(quickActionRequests, allOrNothing)**
Performs the quick actions specified in the quick action request list with the option for partial success, and returns action results.

**Signature**
```java
public static List<QuickAction.QuickActionResult> performQuickActions(List<QuickAction.QuickActionRequest> quickActionRequests, Boolean allOrNothing)
```

**Parameters**
- `quickActionRequests`
  Type: List<QuickAction.QuickActionRequest>
- `allOrNothing`
  Type: Boolean
  Specifies whether this operation allows partial success. If you specify false for this argument and a record fails, the remainder of the DML operation can still succeed. This method returns a result object that can be used to verify which records succeeded, which failed, and why.

**Return Value**
Type: List<QuickAction.QuickActionResult>
Quiddity Enum

Specifies a Quiddity value used by the methods in the System.Request class

Enum Values

The following are the values of the System.Quiddity enum.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANONYMOUS</td>
<td>Execution event is an anonymous Apex block.</td>
</tr>
<tr>
<td>AURA</td>
<td>Execution event is an Aura component.</td>
</tr>
<tr>
<td>BATCH_ACS</td>
<td>Execution event is an API Query Cursor driven batch Apex.</td>
</tr>
<tr>
<td>BATCH_APEX</td>
<td>Execution event is a batch Apex job.</td>
</tr>
<tr>
<td>BATCH_CHUNK_PARALLEL</td>
<td>Execution event is chunks of a batch Apex job running in parallel.</td>
</tr>
<tr>
<td>BATCH_CHUNK_SERIAL</td>
<td>Execution event is chunks of a batch Apex job running in serial.</td>
</tr>
<tr>
<td>BULK_API</td>
<td>Execution event is a bulk API request.</td>
</tr>
<tr>
<td>COMMERCE_INTEGRATION</td>
<td>Execution event is an Apex integration for B2B Commerce.</td>
</tr>
<tr>
<td>DISCOVERABLE_LOGIN</td>
<td>Execution event is Login Discoverable login page used by external users to log in to an Experience Cloud site.</td>
</tr>
<tr>
<td>FUNCTION_CALLBACK</td>
<td>Execution event is a callback function.</td>
</tr>
<tr>
<td>FUTURE</td>
<td>Execution event is a future method.</td>
</tr>
<tr>
<td>INBOUND_EMAIL_SERVICE</td>
<td>Execution event is an Apex inbound email service.</td>
</tr>
<tr>
<td>INVOCABLE_ACTION</td>
<td>Execution event is an invocable action.</td>
</tr>
<tr>
<td>PLATFORM_EVENT_PUBLISH_CALLBACK</td>
<td>Execution event is an Apex publish callback for platform events.</td>
</tr>
<tr>
<td>POST_INSTALL_SCRIPT</td>
<td>Execution event is a managed package install or upgrade.</td>
</tr>
<tr>
<td>QUEUEABLE</td>
<td>Execution event is a queueable Apex operation.</td>
</tr>
<tr>
<td>QUICK_ACTION</td>
<td>Execution event is a quick action.</td>
</tr>
<tr>
<td>REMOTE_ACTION</td>
<td>Execution event is a remote action.</td>
</tr>
<tr>
<td>REST</td>
<td>Execution event is an Apex RESTful Web service.</td>
</tr>
<tr>
<td>RUNTEST_ASYNC</td>
<td>Execution event is Apex tests running asynchronously.</td>
</tr>
<tr>
<td>RUNTEST_DEPLOY</td>
<td>Execution event is Apex tests run during deployment.</td>
</tr>
<tr>
<td>RUNTEST_SYNC</td>
<td>Execution event is Apex tests running synchronously.</td>
</tr>
<tr>
<td>SCHEDULED</td>
<td>Execution event is a scheduled Apex job.</td>
</tr>
<tr>
<td>SOAP</td>
<td>Execution event is an Apex SOAP Web service.</td>
</tr>
<tr>
<td>SYNCHRONOUS</td>
<td>Execution event is a synchronous Apex operation.</td>
</tr>
</tbody>
</table>
RemoteObjectController

Use RemoteObjectController to access the standard Visualforce Remote Objects operations in your Remote Objects override methods.

Namespace

System

Usage

RemoteObjectController is supported only for use within Remote Objects methods. See *Overriding Default Remote Objects Operations* in the Visualforce Developer’s Guide for examples of how to use RemoteObjectController with your Visualforce pages.

RemoteObjectController Methods

The following are methods for RemoteObjectController. All methods are static.

IN THIS SECTION:

  - create(type, fields)
    Create a record in the database.
  - del(type, recordIds)
    Delete records from the database.
  - retrieve(type, fields, criteria)
    Retrieve records from the database.
  - updat(type, recordIds, fields)
    Update records in the database.

create(type, fields)

Create a record in the database.

Signature

```java
public static Map<String, Object> create(String type, Map<String, Object> fields)
```

Parameters

- **type**
  Type: String
The sObject type on which create is being called.

fields
Type: Map<String, Object>
The fields and values to set on the new record.

Return Value
Type: Map<String, Object>
The return value is a map that represents the result of the Remote Objects operation. What is returned depends on the results of the call.

Success
A map that contains a single element with the ID of the record created. For example, `{ id: 'recordId' }`.

Failure
A map that contains a single element with the error message for the overall operation. For example, `{ error: 'errorMessage' }`.

del(type, recordIds)
Delete records from the database.

Signature
public static Map<String, Object> del(String type, List<String> recordIds)

Parameters
type
Type: String
The sObject type on which delete is being called.

recordIds
Type: List<String>
The IDs of the records to be deleted.

Return Value
Type: Map<String, Object>
The return value is a map that represents the result of the Remote Objects operation. What is returned depends on how the method was called and the results of the call.

Single Delete—Success
A map that contains a single element with the ID of the record that was deleted. For example, `{ id: 'recordId' }`.

Batch Delete—Success
A map that contains a single element, an array of Map<String, Object> elements. Each element contains the ID of a record that was deleted and an array of errors, if there were any, for that record’s individual delete. For example, `{ results: [ { id: 'recordId', errors: ['errorMessage'], ... } ], ... ] }.
Single and Batch Delete—Failure

A map that contains a single element with the error message for the overall operation. For example, 

```json
{ error: 'errorMessage' }
```

**retrieve(type, fields, criteria)**

Retrieve records from the database.

**Signature**

```java
public static Map<String,Object> retrieve(String type, List<String> fields, Map<String,Object> criteria)
```

**Parameters**

- **type**
  - Type: String
  - The sObject type on which retrieve is being called.

- **fields**
  - Type: List<String>
  - The fields to retrieve for each record.

- **criteria**
  - Type: Map<String,Object>
  - The criteria to use when performing the query.

**Return Value**

Type: Map<String,Object>

The return value is a map that represents the result of the Remote Objects operation. What is returned depends on the results of the call.

**Success**

A map that contains the following elements.

- **records**: An array of records that match the query conditions.
- **type**: A string that indicates the type of the sObject that was retrieved.
- **size**: The number of records in the response.

**Failure**

A map that contains a single element with the error message for the overall operation. For example, 

```json
{ error: 'errorMessage' }
```

**updat(type, recordIds, fields)**

Update records in the database.
Signature

```java
public static Map<String, Object> updat(String type, List<String> recordIds, Map<String, Object> fields)
```

Parameters

type
Type: String
The sObject type on which update is being called.

recordIds
Type: List<String>
The IDs of the records to be updated.

fields
Type: Map<String, Object>
The fields to update, and the value to update each field with.

Return Value
Type: Map<String, Object>
The return value is a map that represents the result of the Remote Objects operation. What is returned depends on how the method was called and the results of the call.

Single Update—Success
A map that contains a single element with the ID of the record that was updated. For example, `{ id: 'recordId' }`.

Batch Update—Success
A map that contains a single element, an array of Map<String, Object> elements. Each element contains the ID of the record updated and an array of errors, if there were any, for that record's individual update. For example, `{ results: [ { id: 'recordId', errors: ['errorMessage', ...]}, ...] }`.

Single and Batch Update—Failure
A map that contains a single element with the error message for the overall operation. For example, `{ error: 'errorMessage' }`.

Request Class
Contains methods to obtain the request ID and Quiddity value of the current Salesforce request.

Namespace
System

Usage
Use the Request class to detect the current Apex context at runtime. The methods in the Request class obtain a unique request ID and the Quiddity value that represent the current Apex execution type. These values can also be used to correlate with debug and event logs:

- The request ID is universally unique and present in the debug logs that are triggered by the request.
The request ID and Quiddity values are the same as in the event log files of the Apex Execution event type used in Event Monitoring.

Example

This example code shows how to obtain current Apex code context by retrieving the request ID and Quiddity value of the current request.

```java
//Get info about the current request
Request reqInfo = Request.getCurrent();

//Get the identifier for this request, which is universally unique
//Same as requestId in splunk or REQUEST_ID in event monitoring
String currentRequestId = reqInfo.getRequestId();

//Enum representing how Apex is running. e.g. BULK_API vs LIGHTNING
Quiddity currentType = reqInfo.getQuiddity();

//Use this with a switch statement,
//instead of checking System.isFuture() || System.isQueueable() || ...
```

Request Methods

The following are methods for Request.

:getCurrent()

Returns the current Request object that contains the request ID and Quiddity value.

:getQuiddity()

Returns the Quiddity value of the current Request object.

:getRequestId()

Returns the request ID of the current Request object.

:getCurrent()

Returns the current Request object that contains the request ID and Quiddity value.

**Signature**

```
public static System.Request getCurrent()
```

**Return Value**

Type: System.Request

:getQuiddity()

Returns the Quiddity value of the current Request object.
Signature

```java
public System.Quiddity getQuiddity()
```

Return Value

Type: `System.Quiddity`

Uses the values from the Quiddity enum. This value identifies the type of execution event associated with the current request.

```java
getRequestId()
```

Returns the request ID of the current Request object.

Signature

```java
public String getRequestId()
```

Return Value

Type: `String`

### ResetPasswordResult Class

Represents the result of a password reset.

#### Namespace

`System`

#### ResetPasswordResult Methods

The following are instance methods for `ResetPasswordResult`.

**IN THIS SECTION:**

- `getPassword()`: Returns the password generated by the `System.resetPassword` method call.

**getPassword()**

Returns the password generated by the `System.resetPassword` method call.

Signature

```java
public String getPassword()
```

Return Value

Type: `String`
RestContext Class

Contains the RestRequest and RestResponse objects.

Namespace

System

Usage

Use the System.RestContext class to access the RestRequest and RestResponse objects in your Apex REST methods.

Sample

The following example shows how to use RestContext to access the RestRequest and RestResponse objects in an Apex REST method.

```java
@RestResource(urlMapping='/MyRestContextExample/*')
global with sharing class MyRestContextExample {
    @HttpGet
    global static Account doGet() {
        RestRequest req = RestContext.request;
        RestResponse res = RestContext.response;
        String accountId = req.requestURI.substring(req.requestURI.lastIndexOf('/')+1);
        Account result = [SELECT Id, Name, Phone, Website FROM Account WHERE Id = :accountId];
        return result;
    }
}
```

RestContext Properties

The following are properties for RestContext.

IN THIS SECTION:

- request
  - Returns the RestRequest for your Apex REST method.
- response
  - Returns the RestResponse for your Apex REST method.

**request**

Returns the RestRequest for your Apex REST method.

**Signature**

```java
public RestRequest request {get; set;}
```
Property Value
Type: System.RestRequest

**response**
Returns the RestResponse for your Apex REST method.

**Signature**

```java
public RestResponse response {get; set;}
```

Property Value
Type: System.RestResponse

---

**RestRequest Class**

Use the System.RestRequest class to access and pass request data in a RESTful Apex method.

**Namespace**

System

**Usage**
An Apex RESTful Web service method is defined using one of the REST annotations. For more information about Apex RESTful Web service, see Exposing Apex Classes as REST Web Services.

**Example: An Apex Class with REST Annotated Methods**
The following example shows you how to implement the Apex REST API in Apex. This class exposes three methods that each handle a different HTTP request: GET, DELETE, and POST. You can call these annotated methods from a client by issuing HTTP requests.

```java
@RestResource(urlMapping='/Account/*')
global with sharing class MyRestResource {

    @HttpDelete
global static void doGet() {
        RestRequest req = RestContext.request;
        RestResponse res = RestContext.response;
        String accountId = req.requestURI.substring(req.requestURI.lastIndexOf('/')+1);
        Account account = [SELECT Id FROM Account WHERE Id = :accountId];
        delete account;
    }

    @HttpGet
global static Account doGet() {
        RestRequest req = RestContext.request;
        RestResponse res = RestContext.response;
        String accountId = req.requestURI.substring(req.requestURI.lastIndexOf('/'))+1);
```
Account result = [SELECT Id, Name, Phone, Website FROM Account WHERE Id = :accountId];
            return result;
        }
        
        @HttpPost
        global static String doPost(String name, String phone, String website) {
            Account account = new Account();
            account.Name = name;
            account.phone = phone;
            account.website = website;
            insert account;
            return account.Id;
        }

IN THIS SECTION:

  RestRequest Constructors
  RestRequest Properties
  RestRequest Methods

RestRequest Constructors

The following are constructors for RestRequest.

IN THIS SECTION:

  RestRequest()

  Creates a new instance of the System.RestRequest class.

RestRequest ()

Creates a new instance of the System.RestRequest class.

Signature

public RestRequest ()

RestRequest Properties

The following are properties for RestRequest.

Note: While the RestRequest List and Map properties are read-only, their contents are read-write. You can modify them by calling the collection methods directly or you can use the associated RestRequest methods shown in the previous table.
IN THIS SECTION:

headers
Returns the headers that are received by the request.

httpMethod
Returns one of the supported HTTP request methods.

params
Returns the parameters that are received by the request.

remoteAddress
Returns the IP address of the client making the request.

requestBody
Returns or sets the body of the request.

requestURI
Returns or sets everything after the host in the HTTP request string.

resourcePath
Returns the REST resource path for the request.

headers

Returns the headers that are received by the request.

Signature

public Map<String, String> headers {get; set;}

Property Value

Type: Map<String, String>

httpMethod

Returns one of the supported HTTP request methods.

Signature

public String httpMethod {get; set;}

Property Value

Type: String

Possible values returned:

- DELETE
- GET
- HEAD
- PATCH
- POST
• PUT

**params**

Returns the parameters that are received by the request.

**Signature**

```java
public Map<String, String> params {get; set;}
```

**Property Value**

Type: `Map<String, String>`

**remoteAddress**

Returns the IP address of the client making the request.

**Signature**

```java
public String remoteAddress {get; set;}
```

**Property Value**

Type: `String`

**requestBody**

Returns or sets the body of the request.

**Signature**

```java
public Blob requestBody {get; set;}
```

**Property Value**

Type: `Blob`

**Usage**

If the Apex method has no parameters, then Apex REST copies the HTTP request body into the `RestRequest.requestBody` property. If there are parameters, then Apex REST attempts to deserialize the data into those parameters and the data won't be deserialized into the `RestRequest.requestBody` property.

**requestURI**

Returns or sets everything after the host in the HTTP request string.

**Signature**

```java
public String requestURI {get; set;}
```
Property Value
Type: String

Example
For example, if the request string is https://instance.salesforce.com/services/apexrest/Account/ then the requestURI is /Account/.

resourcePath
Returns the REST resource path for the request.

Signature
public String resourcePath {get; set;}

Property Value
Type: String

Example
For example, if the Apex REST class defines a urlMapping of /MyResource/*, the resourcePath property returns /services/apexrest/MyResource/*.

RestRequest Methods
The following are methods for RestRequest. All are instance methods.

Note: At runtime, you typically don’t need to add a header or parameter to the RestRequest object because they are automatically deserialized into the corresponding properties. The following methods are intended for unit testing Apex REST classes. You can use them to add header or parameter values to the RestRequest object without having to recreate the REST method call.

IN THIS SECTION:
- addHeader(name, value)
  Adds a header to the request header map in an Apex test.
- addParameter(name, value)
  Adds a parameter to the request params map in an Apex test.

addHeader(name, value)
Adds a header to the request header map in an Apex test.

Signature
public Void addHeader(String name, String value)
Parameters

name
    Type: String
value
    Type: String

Return Value
Type: Void

Usage
This method is intended for unit testing of Apex REST classes.
The following headers aren't allowed:
• cookie
• set-cookie
• set-cookie2
• content-length
• authorization
If any of these headers are used, an Apex exception is thrown.

addParameter(name, value)
Adds a parameter to the request params map in an Apex test.

Signature
public Void addParameter(String name, String value)

Parameters

name
    Type: String
value
    Type: String

Return Value
Type: Void

Usage
This method is intended for unit testing of Apex REST classes.

RestResponse Class
Represents an object used to pass data from an Apex RESTful Web service method to an HTTP response.
Namespace

System

Usage

Use the `System.RestResponse` class to pass response data from an Apex RESTful web service method that is defined using one of the REST annotations.

IN THIS SECTION:
- RestResponse Constructors
- RestResponse Properties
- RestResponse Methods

RestResponse Constructors

The following are constructors for `RestResponse`.

IN THIS SECTION:
- `RestResponse()`
  - Creates a new instance of the `System.RestResponse` class.

RestResponse() Creates a new instance of the `System.RestResponse` class.

Signature

```java
public RestResponse()
```

RestResponse Properties

The following are properties for `RestResponse`.

**Note:** While the `RestResponse.List` and `Map` properties are read-only, their contents are read-write. You can modify them by calling the collection methods directly or you can use the associated `RestResponse` methods shown in the previous table.

IN THIS SECTION:
- `responseBody`
  - Returns or sets the body of the response.
- `headers`
  - Returns the headers to be sent to the response.
- `statusCode`
  - Returns or sets the response status code.
**responseBody**

Returns or sets the body of the response.

**Signature**

```java
public Blob responseBody {get; set;}
```

**Property Value**

Type: Blob

**Usage**

The response is either the serialized form of the method return value or it's the value of the `responseBody` property based on the following rules:

- If the method returns void, then Apex REST returns the response in the `responseBody` property.
- If the method returns a value, then Apex REST serializes the return value as the response. If the return value contains fields with null value, those fields are not serialized in the response.

**headers**

Returns the headers to be sent to the response.

**Signature**

```java
public Map<String, String> headers {get; set;}
```

**Property Value**

Type: `Map<String, String>`

**statusCode**

Returns or sets the response status code.

**Signature**

```java
public Integer statusCode {get; set;}
```

**Property Value**

Type: Integer

**Status Codes**

The following are valid response status codes. The status code is returned by the `RestResponse.statusCode` property.

- **Note:** If you set the `RestResponse.statusCode` property to a value that's not listed in the table, then an HTTP status of 500 is returned with the error message "Invalid status code for HTTP response: nnn" where nnn is the invalid status code value.
<table>
<thead>
<tr>
<th>Status Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>OK</td>
</tr>
<tr>
<td>201</td>
<td>CREATED</td>
</tr>
<tr>
<td>202</td>
<td>ACCEPTED</td>
</tr>
<tr>
<td>204</td>
<td>NO_CONTENT</td>
</tr>
<tr>
<td>206</td>
<td>PARTIAL_CONTENT</td>
</tr>
<tr>
<td>300</td>
<td>MULTIPLE_CHOICES</td>
</tr>
<tr>
<td>301</td>
<td>MOVED_PERMANENTLY</td>
</tr>
<tr>
<td>302</td>
<td>FOUND</td>
</tr>
<tr>
<td>304</td>
<td>NOT_MODIFIED</td>
</tr>
<tr>
<td>400</td>
<td>BAD_REQUEST</td>
</tr>
<tr>
<td>401</td>
<td>UNAUTHORIZED</td>
</tr>
<tr>
<td>403</td>
<td>FORBIDDEN</td>
</tr>
<tr>
<td>404</td>
<td>NOT_FOUND</td>
</tr>
<tr>
<td>405</td>
<td>METHOD_NOT_ALLOWED</td>
</tr>
<tr>
<td>406</td>
<td>NOT_ACCEPTABLE</td>
</tr>
<tr>
<td>409</td>
<td>CONFLICT</td>
</tr>
<tr>
<td>410</td>
<td>GONE</td>
</tr>
<tr>
<td>412</td>
<td>PRECONDITION_FAILED</td>
</tr>
<tr>
<td>413</td>
<td>REQUEST_ENTITY_TOO_LARGE</td>
</tr>
<tr>
<td>414</td>
<td>REQUEST_URL_TOO_LARGE</td>
</tr>
<tr>
<td>415</td>
<td>UNSUPPORTED_MEDIA_TYPE</td>
</tr>
<tr>
<td>417</td>
<td>EXPECTATION_FAILED</td>
</tr>
<tr>
<td>500</td>
<td>INTERNAL_SERVER_ERROR</td>
</tr>
<tr>
<td>503</td>
<td>SERVER_UNAVAILABLE</td>
</tr>
</tbody>
</table>

**RestResponse Methods**

The following are instance methods for `RestResponse`.

**Note:** At runtime, you typically don’t need to add a header to the `RestResponse` object because it’s automatically deserialized into the corresponding properties. The following methods are intended for unit testing Apex REST classes. You can use them to add header or parameter values to the `RestRequest` object without having to recreate the REST method call.
IN THIS SECTION:

addHeader(name, value)
Adds a header to the response header map.

addHeader(name, value)
Adds a header to the response header map.

**Signature**

```java
public Void addHeader(String name, String value)
```

**Parameters**

- `name`
  Type: `String`

- `value`
  Type: `String`

**Return Value**

Type: `Void`

**Usage**

The following headers aren't allowed:

- cookie
- set-cookie
- set-cookie2
- content-length
- authorization
- Header names that aren't RFC 7230 compliant

If any of these headers are used, an Apex exception is thrown.

---

**SandboxPostCopy Interface**

To make your sandbox environment business ready, automate data manipulation or business logic tasks. Extend this interface and add methods to perform post-copy tasks, then specify the class during sandbox creation.

**Namespace**

`System`

**Usage**

Create an Apex class that implements this interface. Specify your class during sandbox creation. After your sandbox is created, the `runApexClass(context)` method in your class runs using the automated process user's permissions.
**Important:** The SandboxPostCopy Apex class is executed at the end of the sandbox copy using a special Automated Process user that isn’t visible within the org. This user doesn’t have access to all object and features; therefore, the Apex script cannot access all objects and features. If the script fails, run the script after sandbox activation as a user with appropriate permissions.

**IN THIS SECTION:**
- SandboxPostCopy Methods
- SandboxPostCopy Example Implementation

These examples show a simple implementation of the SandboxPostCopy interface and a test for that implementation. To test your SandboxPostCopy implementation, use the `System.Test.testSandboxPostCopyScript()` method.

**SEE ALSO:**
- Tooling API: SandboxInfo
- Tooling API: SandboxInfo

### SandboxPostCopy Methods

The following method is for SandboxPostCopy.

**IN THIS SECTION:**
- `runApexClass(context)`

Executes actions in a new sandbox to prepare it for use. For example, add logic to this method to create users, run sanitizing code on records, and perform other setup tasks.

**runApexClass(context)**

Executes actions in a new sandbox to prepare it for use. For example, add logic to this method to create users, run sanitizing code on records, and perform other setup tasks.

**Signature**

```java
public void runApexClass(System.SandboxContext context)
```

**Parameters**

- `context`
  - Type: System.SandboxContext
  - The org ID, sandbox ID, and sandbox name for your sandbox. To work with these values, reference `context.organizationId()`, `context.sandboxId()`, and `context.sandboxName()` in your code.

**Return Value**

Type: void
SandboxPostCopy Example Implementation

These examples show a simple implementation of the SandboxPostCopy interface and a test for that implementation. To test your SandboxPostCopy implementation, use the `System.Test.testSandboxPostCopyScript()` method.

**Important:** The SandboxPostCopy Apex class is executed at the end of the sandbox copy using a special Automated Process user that isn’t visible within the org. This user doesn’t have access to all object and features; therefore, the Apex script can’t access all objects and features. If the script fails, run the script after sandbox activation as a user with appropriate permissions.

This example implements the `System.SandboxPostCopy` interface.

```java
global class PrepareMySandbox implements SandboxPostCopy {
    global PrepareMySandbox() {
        //Implementations of SandboxPostCopy must have a no-arg constructor.
        //This constructor is used during the sandbox copy process.
        //You can also implement constructors with arguments, but be aware that
        //they won’t be used by the sandbox copy process (unless as part of the
        //no-arg constructor).
        this(some_args);
    }

    global PrepareMySandbox(String some_args) {
        //Logic for constructor.
    }

    global void runApexClass(SandboxContext context) {
        System.debug('Org ID: ' + context.organizationId());
        System.debug('Sandbox ID: ' + context.sandboxId());
        System.debug('Sandbox Name: ' + context.sandboxName());

        // Insert logic here to prepare the sandbox for use.
    }
}
```

The following example tests the implementation using the `System.Test.testSandboxPostCopyScript()` method. This method takes four parameters: a reference to a class that implements the SandboxPostCopy interface, and the three fields on the context object that you pass to the `runApexClass(context)` method. An overload on the method takes an optional Boolean parameter to indicate if the test should be performed as the Automated Process user.

```java
@isTest
class PrepareMySandboxTest {

    @isTest
    static void testMySandboxPrep() {
        // Insert logic here to create records of the objects that the class you’re testing
        // manipulates.
        Test.startTest();

        //Execute test script with RunAsAutoProcUser set to true
        Test.testSandboxPostCopyScript(
            new PrepareMySandbox(), UserInfo.getOrganizationId(),
            ...
        );
    }
}
```
UserInfo.getOrganizationId(), UserInfo.getOrganizationName(), true);

Test.stopTest();

// Insert assert statements here to check that the records you created above have
// the values you expect.

For more information on testing, see Testing Apex.

**Scheduled Interface**

The class that implements this interface can be scheduled to run at different intervals.

**Namespace**

*System*

SEE ALSO:

*Apex Developer Guide: Scheduler*

**Scheduled Methods**

The following are methods for *Scheduled*.

**IN THIS SECTION:**

*execute(context)*

Executes the scheduled Apex job.

```
execute(context)
```

Executes the scheduled Apex job.

**Signature**

```
public Void execute(SchedulableContext context)
```

**Parameters**

*context*

Type: *System.SchedulableContext*

Contains the job ID.

**Return Value**

Type: Void
ScheduledContext Interface

Represents the parameter type of a method in a class that implements the `Scheduled` interface and contains the scheduled job ID. This interface is implemented internally by Apex.

Namespace

`System`

SEE ALSO:

`Scheduled Interface`

ScheduledContext Methods

The following are methods for `ScheduledContext`.

IN THIS SECTION:

`getTriggerId()`

Returns the ID of the CronTrigger scheduled job.

```java
public Id getTriggerId()
```

Return Value

Type: `ID`

Schema Class

Contains methods for obtaining schema describe information.

Namespace

`System`

Schema Methods

The following are methods for `Schema`. All methods are static.
IN THIS SECTION:

getGlobalDescribe()
Returns a map of all sObject names (keys) to sObject tokens (values) for the standard and custom objects defined in your organization.

describeDataCategoryGroups(sObjectName)
Returns a list of the category groups associated with the specified objects.

describeSObjects(sObjectTypes)
Describes metadata (field list and object properties) for the specified sObject or array of sObjects.

describeSObjects(SObjectTypes, SObjectDescribeOptions)
Describes metadata such as field list and object properties for the specified list of SObjects. The default describe option for this method is SObjectDescribeOptions.DEFERRED, which indicates lazy initialization of describe attributes on first use.

describeTabs()
Returns information about the standard and custom apps available to the running user.

describeDataCategoryGroupStructures(pairs, topCategoriesOnly)
Returns available category groups along with their data category structure for objects specified in the request.

globalDescribe()

Returns a map of all sObject names (keys) to sObject tokens (values) for the standard and custom objects defined in your organization.

**Signature**

```
public static Map<String, Schema.SObjectType> getGlobalDescribe()
```

**Return Value**

Type: `Map<String, Schema.SObjectType>`

**Usage**

For more information on accessing SObjects, see Accessing All sObjects.

**Example**

```
Map<String, Schema.SObjectType> gd = Schema.getGlobalDescribe();
```

describeDataCategoryGroups(sObjectName)

Returns a list of the category groups associated with the specified objects.

**Signature**

```
public static List<Schema.DescribeDataCategoryGroupResult> describeDataCategoryGroups(List<String> sObjectNames)
```
Parameters

sObjectName
Type: List<String>

Return Value
Type: List<Schema.DescribeDataCategoryGroupResult>

Usage
You can specify one of the following sObject names:
• KnowledgeArticleVersion—to retrieve category groups associated with article types.
• Question—to retrieve category groups associated with questions.
For more information and code examples using describeDataCategoryGroups, see Accessing All Data Categories Associated with an sObject.
For additional information about articles and questions, see "Work with Articles and Translations" in the Salesforce online help.

describeSObjects (sObjectTypes)
Describes metadata (field list and object properties) for the specified sObject or array of sObjects.

Signature
public static List<Schema.DescribeSObjectResult> describeSObjects(List<String> sObjectTypes)

Parameters

sObjectTypes
Type: List<String>
The sObjectTypes argument is a list of sObject type names you want to describe.

Return Value
Type: List<Schema.DescribeSObjectResult>

Usage
This method is similar to the getDescribe method on the Schema.sObjectType token. Unlike the getDescribe method, this method allows you to specify the sObject type dynamically and describe more than one sObject at a time.
You can first call getGlobalDescribe to retrieve a list of all objects for your organization, then iterate through the list and use describeSObjects to obtain metadata about individual objects.

Example
Schema.DescribeSObjectResult[] descResult = Schema.describeSObjects(new String[]{'Account','Contact'});
**describeSObjects(SObjectTypes, SObjectDescribeOptions)**

Describes metadata such as field list and object properties for the specified list of SObjects. The default describe option for this method is SObjectDescribeOptions.DEFERRED, which indicates lazy initialization of describe attributes on first use.

**Signature**

```java
public static List<Schema.DescribeSObjectResult> describeSObjects(List<String> SObjectTypes, Object SObjectDescribeOptions)
```

**Parameters**

*SObjectTypes*

Type: List<String>

The list of SObject types to describe.

*SObjectDescribeOptions*

Type: Object

The effective describe option used for the SObject.

**Return Value**

Type: List<Schema.DescribeSObjectResult>

**describeTabs()**

Returns information about the standard and custom apps available to the running user.

**Signature**

```java
public static List<Schema.DescribeTabSetResult> describeTabs()
```

**Return Value**

Type: List<Schema.DescribeTabSetResult>

**Usage**

An app is a group of tabs that works as a unit to provide application functionality. For example, two of the standard Salesforce apps are “Sales” and “Service.”

The `describeTabs` method returns the minimum required metadata that can be used to render apps in another user interface. Typically, this call is used by partner applications to render Salesforce data in another user interface, such as in a mobile or connected app.

In the Salesforce user interface, users have access to standard apps (and can also have access to custom apps) as listed in the Salesforce app menu at the top of the page. Selecting an app name in the menu allows the user to switch between the listed apps at any time.

**Note:** The “All Tabs” tab isn’t included in the list of described tabs.
Example

This example shows how to call the describeTabs method.

```java
Schema.DescribeTabSetResult[] tabSetDesc = Schema.describeTabs();
```

This longer example shows how to obtain describe metadata information for the Sales app. For each tab, the example gets describe information, such as the icon URL, whether the tab is custom or not, and colors. The describe information is written to the debug output.

```java
// Get tab set describes for each app
List<Schema.DescribeTabSetResult> tabSetDesc = Schema.describeTabs();

// Iterate through each tab set describe for each app and display the info
for (DescribeTabSetResult tsr : tabSetDesc) {
    String appLabel = tsr.getLabel();
    System.debug('Label: ' + appLabel);
    System.debug('Logo URL: ' + tsr.getLogoUrl());
    System.debug('isSelected: ' + tsr.isSelected());
    String ns = tsr.getNamespace();
    if (ns == '') {
        System.debug('The ' + appLabel + ' app has no namespace defined.');
    } else {
        System.debug('Namespace: ' + ns);
    }

    // Display tab info for the Sales app
    if (appLabel == 'Sales') {
        List<Schema.DescribeTabResult> tabDesc = tsr.getTabs();
        System.debug('-- Tab information for the Sales app --');
        for (Schema.DescribeTabResult tr : tabDesc) {
            System.debug('getLabel: ' + tr.getLabel());
            System.debug('getColors: ' + tr.getColors());
            System.debug('getIconUrl: ' + tr.getIconUrl());
            System.debug('getIcons: ' + tr.getIcons());
            System.debug('getMiniIconUrl: ' + tr.getMiniIconUrl());
            System.debug('getSobjectName: ' + tr.getSobjectName());
            System.debug('getUrl: ' + tr.getUrl());
            System.debug('isCustom: ' + tr.isCustom());
        }
    }
}
```

// Example debug statement output
// DEBUG|Label: Sales
// DEBUG|Logo URL: https://MyDomainName.my.salesforce.com/img/seasonLogos/2014_winter_aloha.png
// DEBUG|isSelected: true
// DEBUG|The Sales app has no namespace defined.// DEBUG|-- Tab information for the Sales app --
// (This is an example debug output for the Accounts tab.)
// DEBUG|getLabel: Accounts
// DEBUG|getColors:
// (Schema.DescribeColorResult[getColor=236FBD;getContext=primary;getTheme=theme4;],
// Schema.DescribeColorResult[getColor=236FBD;getContext=primary;getTheme=theme3;],
// ...)
describeDataCategoryGroupStructures (pairs, topCategoriesOnly)

Returns available category groups along with their data category structure for objects specified in the request.

Signature

public static List<Schema.DescribeDataCategoryGroupStructureResult> describeDataCategoryGroupStructures(List<Schema.DataCategoryGroupSobjectTypePair> pairs, Boolean topCategoriesOnly)

Parameters

pairs

Type: List<Schema.DataCategoryGroupSobjectTypePair>

The pairs argument is one or more category groups and objects to query Schema.DataCategoryGroupSobjectTypePairs. Visible data categories are retrieved for the specified object. For more information on data category group visibility, see “Data Category Visibility” in Salesforce Help.

topCategoriesOnly

Type: Boolean

Use true to return only the top visible category and false to return all the visible categories, depending on the user’s data category group visibility settings. For more information on data category group visibility, see Data Category Visibility in Salesforce Help.

Return Value

Type: List<Schema.DescribeDataCategoryGroupStructureResult>

Search Class

Use the methods of the Search class to perform dynamic SOSL queries.

Namespace

System
Search Methods

The following are static methods for Search.

IN THIS SECTION:

find(searchQuery)
Performs a dynamic SOSL query that can include the SOSL WITH SNIPPET clause. Snippets provide more context for users in Salesforce Knowledge article search results.

find(searchQuery, accessLevel)
Performs a dynamic SOSL query that can include the SOSL WITH SNIPPET clause. Snippets provide more context for users in Salesforce Knowledge article search results.

query(query)
Performs a dynamic SOSL query.

query(query, accessLevel)
Performs a dynamic SOSL query.

suggest(searchQuery, sObjectType, suggestions)
Returns a list of records or Salesforce Knowledge articles whose names or titles match the user’s search query string. Use this method to provide users with shortcuts to navigate to relevant records or articles before they perform a search.

suggest(searchQuery, sObjectType, suggestions, accessLevel)
Returns a list of records or Salesforce Knowledge articles whose names or titles match the user’s search query string. Use this method to provide users with shortcuts to navigate to relevant records or articles before they perform a search.

find(searchQuery)
Performs a dynamic SOSL query that can include the SOSL WITH SNIPPET clause. Snippets provide more context for users in Salesforce Knowledge article search results.

Signature

public static Search.SearchResults find(String searchQuery)

Parameters

searchQuery
Type: String
A SOSL query string.

Return Value

Type: Search.SearchResults

Usage

Use this method wherever a static SOSL query can be used, such as in regular assignment statements and for loops.
See Use Dynamic SOSL to Return Snippets.

SEE ALSO:
get(sObjectType)
Apex Developer Guide: Dynamic SOSL

**find(searchQuery, accessLevel)**

Performs a dynamic SOSL query that can include the SOSL WITH SNIPPET clause. Snippets provide more context for users in Salesforce Knowledge article search results.

**Signature**

```java
public static Search.SearchResults find(String searchQuery, System.AccessLevel accessLevel)
```

**Parameters**

- **searchQuery**
  - Type: String
  - A SOSL query string.
- **accessLevel**
  - Type: System.AccessLevel
  - (Optional) The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

**Return Value**

- Type: Search.SearchResults

**Usage**

Use this method wherever a static SOSL query can be used, such as in regular assignment statements and for loops.
See Use Dynamic SOSL to Return Snippets.

**query(query)**

Performs a dynamic SOSL query.

**Signature**

```java
public static sObject[] query(String query)
```
Parameters

query
  Type: String
  A SOSL query string.
  To create a SOSL query that includes the WITH SNIPPET clause, use the Search.find(String searchQuery) method instead.

Return Value
Type: sObject[sObject[]]

Usage
This method can be used wherever a static SOSL query can be used, such as in regular assignment statements and for loops.
For more information, see Dynamic SOSL.

query(query, accessLevel)
Performs a dynamic SOSL query.

Signature
public static List<List<SObject>> query(String query, System.AccessLevel accessLevel)

Parameters
query
  Type: String
  A SOSL query string.
  To create a SOSL query that includes the WITH SNIPPET clause, use the Search.find(String searchQuery) method instead.

accessLevel
  Type: System.AccessLevel
  (Optional) The accessLevel parameter specifies whether the method runs in system mode (AccessLevel.SYSTEM_MODE) or user mode (AccessLevel.USER_MODE). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

Return Value
Type: sObject[sObject[]]

Usage
This method can be used wherever a static SOSL query can be used, such as in regular assignment statements and for loops.
For more information, see Dynamic SOSL.
suggest(searchQuery, sObjectType, suggestions)

Returns a list of records or Salesforce Knowledge articles whose names or titles match the user's search query string. Use this method to provide users with shortcuts to navigate to relevant records or articles before they perform a search.

Signature

public static Search.SuggestionResults suggest(String searchQuery, String sObjectType, Search.SuggestionOption suggestions)

Parameters

searchQuery
  Type: String
  A SOSL query string.

sObjectType
  Type: String
  An sObject type.

options
  Type: Search.SuggestionOption
  This object contains options that change the suggestion results.

  If the searchQuery returns KnowledgeArticleVersion objects, pass an options parameter with a Search.SuggestionOption object that contains a language KnowledgeSuggestionFilter and a publish status KnowledgeSuggestionFilter.

  For suggestions for all other record types, the only supported option is a limit, which sets the maximum number of suggestions returned.

Return Value

Type: SuggestionResults

Usage

Use this method to return:

Suggestions for Salesforce Knowledge articles (KnowledgeArticleVersion)

  Salesforce Knowledge must be enabled in your organization. The user must have the “View Articles” permission enabled.

  The articles suggested include only the articles the user can access, based on the data categories and article types the user has permissions to view.

Suggestions for other record types

  The records suggested include only the records the user can access.

  This method returns a record if its name field starts with the text in the search string. This method automatically appends an asterisk wildcard (*) at the end of the search string. Records that contain the search string within a word aren’t considered a match.

  Records are suggested if the entire search string is found in the record name, in the same order as specified in the search string. For example, the text string national u is treated as national u* and returns “National Utility” and “National Urban Company” but not “National Company Utility” or “Urban National Company”.

3327
Note: If the user’s search query contains quotation marks or wildcards, those symbols are automatically removed from the query string in the URI.

SEE ALSO:

* Apex Developer Guide: Suggest Salesforce Knowledge Articles*

**suggest**(searchQuery, sObjectType, suggestions, accessLevel)

Returns a list of records or Salesforce Knowledge articles whose names or titles match the user’s search query string. Use this method to provide users with shortcuts to navigate to relevant records or articles before they perform a search.

**Signature**

```java
public static Search.SuggestionResults suggest(String searchQuery, String sObjectType, Search.SuggestionOption suggestions, System.AccessLevel accessLevel)
```

**Parameters**

- **searchQuery**
  - Type: `String`
  - A SOSL query string.

- **sObjectType**
  - Type: `String`
  - An sObject type.

- **suggestions**
  - Type: `Search.SuggestionOption`
  - This object contains options that change the suggestion results.
  - If the `searchQuery` returns KnowledgeArticleVersion objects, pass an `options` parameter with a Search.SuggestionOption object that contains a language KnowledgeSuggestionFilter and a publish status KnowledgeSuggestionFilter.
  - For suggestions for all other record types, the only supported option is a limit, which sets the maximum number of suggestions returned.

- **accessLevel**
  - Type: `System.AccessLevel`
  - (Optional) The `accessLevel` parameter specifies whether the method runs in system mode (`AccessLevel.SYSTEM_MODE`) or user mode (`AccessLevel.USER_MODE`). In system mode, the object and field-level permissions of the current user are ignored, and the record sharing rules are controlled by the class sharing keywords. In user mode, the object permissions, field-level security, and sharing rules of the current user are enforced. System mode is the default.

**Return Value**

- Type: `SuggestionResults`

**Usage**

Use this method to return:
Suggestions for Salesforce Knowledge articles (KnowledgeArticleVersion)

Salesforce Knowledge must be enabled in your organization. The user must have the “View Articles” permission enabled.

The articles suggested include only the articles the user can access, based on the data categories and article types the user has permissions to view.

Suggestions for other record types

The records suggested include only the records the user can access.

This method returns a record if its name field starts with the text in the search string. This method automatically appends an asterisk wildcard (*) at the end of the search string. Records that contain the search string within a word aren’t considered a match.

Records are suggested if the entire search string is found in the record name, in the same order as specified in the search string. For example, the text string national u is treated as national u* and returns “National Utility” and “National Urban Company” but not “National Company Utility” or “Urban National Company”.

Note: If the user’s search query contains quotation marks or wildcards, those symbols are automatically removed from the query string in the URI.

Security Class

Contains methods to securely implement Apex applications.

Namespace

System

Usage

In the context of the current user’s create, read, update, or upsert access permission, use the Security class methods to:

- Strip fields that aren’t visible from query and subquery results
- Remove inaccessible fields before a DML operation without causing an exception
- Sanitize SObjects that have been deserialized from an untrusted source

IN THIS SECTION:

Security Methods

Security Methods

The following are methods for Security.

IN THIS SECTION:

stripInaccessible(accessCheckType, sourceRecords, enforceRootObjectCRUD)

Creates a list of sObjects from the source records, which are stripped of fields that fail the field-level security checks for the current user. The method also provides an option to enforce an object-level access check.

stripInaccessible(accessCheckType, sourceRecords)

Creates a list of sObjects from the source records, which are stripped of fields that fail the field-level security checks for the current user.
stripInaccessible(accessCheckType, sourceRecords, enforceRootObjectCRUD)

Creates a list of sObjects from the source records, which are stripped of fields that fail field-level and object-level access checks. Apex enforces field-level security (FLS) and object permissions as per the specified permission set, in addition to the running user’s permissions.

**stripInaccessible(accessCheckType, sourceRecords, enforceRootObjectCRUD)**

Creates a list of sObjects from the source records, which are stripped of fields that fail the field-level security checks for the current user. The method also provides an option to enforce an object-level access check.

**Signature**

```java
public static System.SObjectAccessDecision stripInaccessible(System.AccessType accessCheckType, List<SObject> sourceRecords, Boolean enforceRootObjectCRUD)
```

**Parameters**

- **accessCheckType**
  
  Type: `System.AccessType`

  Uses values from the `AccessType` enum. This parameter determines the type of field-level access check to be performed. To check the current user's field-level access, use the `Schema.DescribeFieldResult` methods — `isCreatable()`, `isAccessible()`, or `isUpdatable()`.

- **sourceRecords**
  
  Type: `List<SObject>`

  A list of sObjects to be checked for fields that aren’t accessible in the context of the current user’s operation.

- **enforceRootObjectCRUD**
  
  Type: `Boolean`

  Indicates whether an object-level access check is performed. If this parameter is set to `true` and the access check fails, the method throws an exception. The default value of this optional parameter is `true`.

**Return Value**

Type: `System.SObjectAccessDecision`

**Example**

In this example, the user doesn’t have permission to create the `Probability` field of an Opportunity.

```java
List<Opportunity> opportunities = new List<Opportunity>(
    new Opportunity(Name='Opportunity1'),
    new Opportunity(Name='Opportunity2', Probability=95)
);

// Strip fields that are not creatable
SObjectAccessDecision decision = Security.stripInaccessible(
    AccessType.CREATABLE,
    opportunities);

// Print stripped records
```
for (SObject strippedOpportunity : decision.getRecords()) {
    System.debug(strippedOpportunity);
}

// Print modified indexes
System.debug(decision.getModifiedIndexes());

// Print removed fields
System.debug(decision.getRemovedFields());

//Lines from output log
///|DEBUG|Opportunity:{Name=Opportunity1}
///|DEBUG|Opportunity:{Name=Opportunity2}
///|DEBUG|{1}
///|DEBUG|{Opportunity={Probability}}

stripInaccessible(accessCheckType, sourceRecords)

Creates a list of sObjects from the source records, which are stripped of fields that fail the field-level security checks for the current user.

**Signature**

public static System.SObjectAccessDecision stripInaccessible(System.AccessType accessCheckType, List<SObject> sourceRecords)

**Parameters**

*accessCheckType*
- Type: System.AccessType
  - Uses values from the AccessType enum. This parameter determines the type of field-level access check to be performed. To check the current user's field-level access, use the Schema.DescribeFieldResult methods — isCreatable(), isAccessible(), or isUpdatable().

*sourceRecords*
- Type: List<SObject>
  - A list of sObjects to be checked for fields that aren't accessible in the context of the current user's operation.

**Return Value**

Type: System.SObjectAccessDecision

**Example**

In this example, the user doesn't have permission to read the ActualCost field of a Campaign.

List<Campaign> campaigns = new List<Campaign>{
    new Campaign(Name='Campaign1', BudgetedCost=1000, ActualCost=2000),
    new Campaign(Name='Campaign2', BudgetedCost=4000, ActualCost=1500)
};
insert campaigns;

// Strip fields that are not readable
SObjectAccessDecision decision = Security.stripInaccessible(
    AccessType.READABLE,
    [SELECT Name, BudgetedCost, ActualCost from Campaign]);

// Print stripped records
for (SObject strippedCampaign : decision.getRecords()) {
    System.debug(strippedCampaign); // Does not display ActualCost
}

// Print modified indexes
System.debug(decision.getModifiedIndexes());

// Print removed fields
System.debug(decision.getRemovedFields());

//Lines from output log
///|DEBUG|Campaign:{Name=Campaign1, BudgetedCost=1000, Id=701xx00000011nhAAA}
///|DEBUG|Campaign:{Name=Campaign2, BudgetedCost=4000, Id=701xx00000011niAAA}
///|DEBUG|{0, 1}
///|DEBUG|{ActualCost}

stripInaccessible(accessCheckType, sourceRecords, enforceRootObjectCRUD, permissionSetId)(Developer Preview)

Creates a list of sObjects from the source records, which are stripped of fields that fail field-level and object-level access checks. Apex enforces field-level security (FLS) and object permissions as per the specified permission set, in addition to the running user's permissions.

Note: Feature is available as a developer preview. Feature isn’t generally available unless or until Salesforce announces its general availability in documentation or in press releases or public statements. All commands, parameters, and other features are subject to change or deprecation at any time, with or without notice. Don’t implement functionality developed with these commands or tools in a production environment. You can provide feedback and suggestions for the “Permission Sets with User Mode” feature in the Trailblazer Community.

This feature is available in scratch orgs where the ApexUserModeWithPermset feature is enabled. If the feature isn’t enabled, Apex code with this feature can be compiled but not executed.

Signature

```
public static System.SObjectAccessDecision stripInaccessible(System.AccessType accessCheckType, List<SObject> sourceRecords, Boolean enforceRootObjectCRUD, Id permissionSetId)
```

Parameters

accessCheckType
Type: System.AccessType

Uses values from the AccessType enum. This parameter determines the type of field-level access check to be performed. To check the current user’s field-level access, use the Schema.DescribeFieldResult methods — isCreatable(), isAccessible(), or isUpdatable().

sourceRecords
Type: List<SObject>
enforceRootObjectCRUD
Type: Boolean
Indicates whether an object-level access check is performed. If this parameter is set to true and the access check fails, the method throws an exception. The default value of this optional parameter is true.

permissionSetId
Type: Id
Permissions in the specified permission set are enforced in addition to the running user’s permissions.

Return Value
Type: System.SObjectAccessDecision

SelectOption Class
A SelectOption object specifies one of the possible values for a Visualforce selectCheckboxes, selectList, or selectRadio component.

Namespace
System
SelectOption consists of a label that is displayed to the end user, and a value that is returned to the controller if the option is selected. A SelectOption can also be displayed in a disabled state, so that a user cannot select it as an option, but can still view it.

Instantiation
In a custom controller or controller extension, you can instantiate a SelectOption in one of the following ways:

* `SelectOption option = new SelectOption(value, label, isDisabled);`

  where value is the String that is returned to the controller if the option is selected by a user, label is the String that is displayed to the user as the option choice, and isDisabled is a Boolean that, if true, specifies that the user cannot select the option, but can still view it.

* `SelectOption option = new SelectOption(value, label);`

  where value is the String that is returned to the controller if the option is selected by a user, and label is the String that is displayed to the user as the option choice. Because a value for isDisabled is not specified, the user can both view and select the option.

Example
The following example shows how a list of SelectOptions objects can be used to provide possible values for a selectCheckboxes component on a Visualforce page. In the following custom controller, the getItems method defines and returns the list of possible SelectOption objects:

public class sampleCon {
String[] countries = new String[]{};

public PageReference test() {
    return null;
}

public List<SelectOption> getItems() {
    List<SelectOption> options = new List<SelectOption>();
    options.add(new SelectOption('US', 'US'));
    options.add(new SelectOption('CANADA', 'Canada'));
    options.add(new SelectOption('MEXICO', 'Mexico'));
    return options;
}

public String[] getCountries() {
    return countries;
}

public void setCountries(String[] countries) {
    this.countries = countries;
}

In the following page markup, the <apex:selectOptions> tag uses the getItems method from the controller above to retrieve the list of possible values. Because <apex:selectOptions> is a child of the <apex:selectCheckboxes> tag, the options are displayed as checkboxes:

```apex
<apex:page controller="sampleCon">
    <apex:form>
        <apex:selectCheckboxes value="{!countries}">
            <apex:selectOptions value="{!items}"/>
        </apex:selectCheckboxes>
        <apex:commandButton value="Test" action="{!test}" rerender="out" status="status"/>
    </apex:form>
    <apex:outputPanel id="out">
        <apex:actionstatus id="status" startText="testing...">
            <apex:facet name="stop">
                <apex:outputPanel>
                    <p>You have selected:</p>
                    <apex:datalist value="{!countries}" var="c">{!c}</apex:datalist>
                </apex:outputPanel>
            </apex:facet>
        </apex:actionstatus>
    </apex:outputPanel>
</apex:page>
```

IN THIS SECTION:

SelectOption Constructors
SelectOption Methods
SelectOption Constructors

The following are constructors for SelectOption.

IN THIS SECTION:

SelectOption(value, label)
Creates a new instance of the SelectOption class using the specified value and label.

SelectOption(value, label, isDisabled)
Creates a new instance of the SelectOption class using the specified value, label, and disabled setting.

SelectOption(value, label)
Creates a new instance of the SelectOption class using the specified value and label.

Signature

public SelectOption(String value, String label)

Parameters

value
  Type: String
  The string that is returned to the Visualforce controller if the option is selected by a user.

label
  Type: String
  The string that is displayed to the user as the option choice.

SelectOption(value, label, isDisabled)
Creates a new instance of the SelectOption class using the specified value, label, and disabled setting.

Signature

public SelectOption(String value, String label, Boolean isDisabled)

Parameters

value
  Type: String
  The string that is returned to the Visualforce controller if the option is selected by a user.

label
  Type: String
  The string that is displayed to the user as the option choice.

isDisabled
  Type: Boolean
  If set to true, the option can’t be selected by the user but can still be viewed.
SelectOption Methods

The following are methods for SelectOption. All are instance methods.

IN THIS SECTION:

getDisabled()
Returns the current value of the SelectOption object’s isDisabled attribute.

getEscapeItem()
Returns the current value of the SelectOption object’s itemEscaped attribute.

getLabel()
Returns the option label that is displayed to the user.

getValue()
Returns the option value that is returned to the controller if a user selects the option.

setDisabled(isDisabled)
Sets the value of the SelectOption object’s isDisabled attribute.

setEscapeItem(itemsEscaped)
Sets the value of the SelectOption object’s itemEscaped attribute.

setLabel(label)
Sets the value of the option label that is displayed to the user.

setValue(value)
Sets the value of the option value that is returned to the controller if a user selects the option.

getDisabled()
Returns the current value of the SelectOption object’s isDisabled attribute.

Signature

public Boolean getDisabled()

Return Value

Type: Boolean

Usage

If isDisabled is set to true, the user can view the option, but cannot select it. If isDisabled is set to false, the user can both view and select the option.

getEscapeItem()
Returns the current value of the SelectOption object’s itemEscaped attribute.

Signature

public Boolean getEscapeItem()
Return Value
Type: Boolean

Usage
If `itemEscaped` is set to `true`, sensitive HTML and XML characters are escaped in the HTML output generated by this component. If `itemEscaped` is set to `false`, items are rendered as written.

`getLabel()`
Returns the option label that is displayed to the user.

Signature
```
public String getLabel()
```

Return Value
Type: String

`getValue()`
Returns the option value that is returned to the controller if a user selects the option.

Signature
```
public String getValue()
```

Return Value
Type: String

`setDisabled(isDisabled)`
Sets the value of the SelectOption object's `isDisabled` attribute.

Signature
```
public Void setDisabled(Boolean isDisabled)
```

Parameters
`isDisabled`
Type: Boolean

Return Value
Type: Void
Usage
If `isDisabled` is set to `true`, the user can view the option, but cannot select it. If `isDisabled` is set to `false`, the user can both view and select the option.

`setEscapeItem(itemsEscaped)`
Sets the value of the SelectOption object's `itemEscaped` attribute.

Signature
```java
public Void setEscapeItem(Boolean itemsEscaped)
```

Parameters
- `itemsEscaped`
  Type: `Boolean`

Return Value
Type: Void

Usage
If `itemEscaped` is set to `true`, sensitive HTML and XML characters are escaped in the HTML output generated by this component. If `itemEscaped` is set to `false`, items are rendered as written.

`setLabel(label)`
Sets the value of the option label that is displayed to the user.

Signature
```java
public Void setLabel(String label)
```

Parameters
- `label`
  Type: `String`

Return Value
Type: Void

`setValue(value)`
Sets the value of the option value that is returned to the controller if a user selects the option.

Signature
```java
public Void setValue(String value)
```
Parameters

value
Type: String

Return Value
Type: Void

Set Class

Represents a collection of unique elements with no duplicate values.

Namespace

System

Usage

The Set methods work on a set, that is, an unordered collection of elements that was initialized using the set keyword. Set elements can be of any data type—primitive types, collections, sObjects, user-defined types, and built-in Apex types. Set methods are all instance methods, that is, they all operate on a particular instance of a Set. The following are the instance methods for sets.

Note:

- Uniqueness of set elements of user-defined types is determined by the equals and hashCode methods, which you provide in your classes. Uniqueness of all other non-primitive types is determined by comparing the objects' fields.
- If the set contains String elements, the elements are case-sensitive. Two set elements that differ only by case are considered distinct.

For more information on sets, see Sets.

IN THIS SECTION:

Set Constructors

Set Methods

Set Constructors

The following are constructors for Set.

IN THIS SECTION:

Set<T>()

Creates a new instance of the Set class. A set can hold elements of any data type T.

Set<T>(setToCopy)

Creates a new instance of the Set class by copying the elements of the specified set. T is the data type of the elements in both sets and can be any data type.
Set<T>(listToCopy)
Creates a new instance of the Set class by copying the list elements. T is the data type of the elements in the set and list and can be any data type.

Set<T>()
Creates a new instance of the Set class. A set can hold elements of any data type T.

Signature
public Set<T>()

Example
// Create a set of strings
Set<String> s1 = new Set<String>();
// Add two strings to it
s1.add('item1');
s1.add('item2');

Set<T>(setToCopy)
Creates a new instance of the Set class by copying the elements of the specified set. T is the data type of the elements in both sets and can be any data type.

Signature
public Set<T>(Set<T> setToCopy)

Parameters
setToCopy
Type: Set<T>
The set to initialize this set with.

Example
Set<String> s1 = new Set<String>();
s1.add('item1');
s1.add('item2');
Set<String> s2 = new Set<String>(s1);
// The set elements in s2 are copied from s1
System.debug(s2);

Set<T>(listToCopy)
Creates a new instance of the Set class by copying the list elements. T is the data type of the elements in the set and list and can be any data type.
Signature

```java
public Set<T>(List<T> listToCopy)
```

Parameters

`listToCopy`

Type: `Integer`

The list to copy the elements of into this set.

Example

```java
List<Integer> ls = new List<Integer>();
ls.add(1);
ls.add(2);
// Create a set based on a list
Set<Integer> s1 = new Set<Integer>(ls);
// Elements are copied from the list to this set
System.debug(s1);// DEBUG|{1, 2}
```

Set Methods

The following are methods for `Set`. All are instance methods.

**IN THIS SECTION:**

- `add(setElement)`
  Adds an element to the set if it is not already present.
- `addAll(fromList)`
  Adds all of the elements in the specified list to the set if they are not already present.
- `addAll(fromSet)`
  Adds all of the elements in the specified set to the set that calls the method if they are not already present.
- `clear()`
  Removes all of the elements from the set.
- `clone()`
  Makes a duplicate copy of the set.
- `contains(setElement)`
  Returns `true` if the set contains the specified element.
- `containsAll(listToCompare)`
  Returns `true` if the set contains all of the elements in the specified list. The list must be of the same type as the set that calls the method.
- `containsAll(setToCompare)`
  Returns `true` if the set contains all of the elements in the specified set. The specified set must be of the same type as the original set that calls the method.
- `equals(set2)`
  Compares this set with the specified set and returns `true` if both sets are equal; otherwise, returns `false`.
hashCode()
Returns the hashcode corresponding to this set and its contents.

isEmpty()
Returns true if the set has zero elements.

remove(setElement)
Removes the specified element from the set if it is present.

removeAll(listOfElementsToRemove)
Removes the elements in the specified list from the set if they are present.

removeAll(setOfElementsToRemove)
Removes the elements in the specified set from the original set if they are present.

retainAll(listOfElementsToRetain)
Retains only the elements in this set that are contained in the specified list.

retainAll(setOfElementsToRetain)
Retains only the elements in the original set that are contained in the specified set.

size()
Returns the number of elements in the set (its cardinality).

toString()
Returns the string representation of the set.

add(setElement)
Adds an element to the set if it is not already present.

**Signature**

```java
public Boolean add(Object setElement)
```

**Parameters**

setElement
Type: Object

**Return Value**

Type: Boolean

**Usage**

This method returns true if the original set changed as a result of the call. For example:

```java
Set<String> myString = new Set<String>{'a', 'b', 'c'};
Boolean result = myString.add('d');
System.assertEquals(true, result);
```

addAll(fromList)
Adds all of the elements in the specified list to the set if they are not already present.
**Signature**

public Boolean addAll(List<Object> fromList)

**Parameters**

*fromList*

  Type: List

**Return Value**

Type: Boolean

Returns true if the original set changed as a result of the call.

**Usage**

This method results in the union of the list and the set. The list must be of the same type as the set that calls the method.

**addAll(fromSet)**

Adds all of the elements in the specified set to the set that calls the method if they are not already present.

**Signature**

public Boolean addAll(Set<Object> fromSet)

**Parameters**

*fromSet*

  Type: Set<Object>

**Return Value**

Type: Boolean

This method returns true if the original set changed as a result of the call.

**Usage**

This method results in the union of the two sets. The specified set must be of the same type as the original set that calls the method.

**Example**

```java
Set<String> myString = new Set<String>{'a', 'b'};
Set<String> sString = new Set<String>{'c'};
Boolean result1 = myString.addAll(sString);
System.assertEquals(true, result1);
```

**clear()**

Removes all of the elements from the set.
Signature
public Void clear()

Return Value
Type: Void

close()
Makes a duplicate copy of the set.

Signature
public Set<Object> close()

Return Value
Type: Set (of same type)

contains(setElement)
Returns true if the set contains the specified element.

Signature
public Boolean contains(Object setElement)

Parameters
setElement
Type: Object

Return Value
Type: Boolean

Example
Set<String> myString = new Set<String>{'a', 'b'};
Boolean result = myString.contains('z');
System.assertEquals(false, result);

containsAll(listToCompare)
Returns true if the set contains all of the elements in the specified list. The list must be of the same type as the set that calls the method.

Signature
public Boolean containsAll(List<Object> listToCompare)
Parameters

listToCompare
Type: List<Object>

Return Value
Type: Boolean

containsAll(setToCompare)
Returns true if the set contains all of the elements in the specified set. The specified set must be of the same type as the original set that calls the method.

Signature
public Boolean containsAll(Set<Object> setToCompare)

Parameters

setToCompare
Type: Set<Object>

Return Value
Type: Boolean

Example

```java
Set<String> myString = new Set<String>{'a', 'b'};
Set<String> sString = new Set<String>{'c'};
Set<String> rString = new Set<String>{'a', 'b', 'c'};

Boolean result1, result2;
result1 = myString.addAll(sString);
system.assertEquals(true, result1);

result2 = myString.containsAll(rString);
System.assertEquals(true, result2);
```

equals(set2)
Compares this set with the specified set and returns true if both sets are equal; otherwise, returns false.

Signature
public Boolean equals(Set<Object> set2)

Parameters

set2
Type: Set<Object>
The `set2` argument is the set to compare this set with.

**Return Value**
Type: Boolean

**Usage**
Two sets are equal if their elements are equal, regardless of their order. The `==` operator is used to compare the elements of the sets. The `==` operator is equivalent to calling the `equals` method, so you can call `set1.equals(set2);` instead of `set1 == set2;`.

**hashCode()**
Returns the hashcode corresponding to this set and its contents.

**Signature**
```
public Integer hashCode()
```

**Return Value**
Type: Integer

**isEmpty()**
Returns `true` if the set has zero elements.

**Signature**
```
public Boolean isEmpty()
```

**Return Value**
Type: Boolean

**Example**
```java
Set<Integer> mySet = new Set<Integer>();
Boolean result = mySet.isEmpty();
System.assertEquals(true, result);
```

**remove(setElement)**
Removes the specified element from the set if it is present.

**Signature**
```
public Boolean remove(Object setElement)
```
Parameters

setElement
Type: Object

Return Value
Type: Boolean
Returns true if the original set changed as a result of the call.

removeAll(listOfElementsToRemove)
Removes the elements in the specified list from the set if they are present.

Signature
public Boolean removeAll(List<Object> listOfElementsToRemove)

Parameters

listOfElementsToRemove
Type: List<Object>

Return Value
Type: Boolean
Returns true if the original set changed as a result of the call.

Usage
This method results in the relative complement of the two sets. The list must be of the same type as the set that calls the method.

Example
```
Set<Integer> mySet = new Set<Integer>{1, 2, 3};
List<Integer> myList = new List<Integer>{1, 3};
Boolean result = mySet.removeAll(myList);
System.assertEquals(true, result);
Integer result2 = mySet.size();
System.assertEquals(1, result2);
```

removeAll(setOfElementsToRemove)
Removes the elements in the specified set from the original set if they are present.

Signature
public Boolean removeAll(Set<Object> setOfElementsToRemove)
Parameters

setOfElementsToRemove

Type: Set<Object>

Return Value

Type: Boolean

This method returns true if the original set changed as a result of the call.

Usage

This method results in the relative complement of the two sets. The specified set must be of the same type as the original set that calls the method.

retainAll(listOfElementsToRetain)

Retains only the elements in this set that are contained in the specified list.

Signature

public Boolean retainAll(List<Object> listOfElementsToRetain)

Parameters

listOfElementsToRetain

Type: List<Object>

Return Value

Type: Boolean

This method returns true if the original set changed as a result of the call.

Usage

This method results in the intersection of the list and the set. The list must be of the same type as the set that calls the method.

Example

Set<Integer> mySet = new Set<Integer>{1, 2, 3};
List<Integer> myList = new List<Integer>{1, 3};
Boolean result = mySet.retainAll(myList);
System.assertEquals(true, result);

retainAll(setOfElementsToRetain)

Retains only the elements in the original set that are contained in the specified set.
Signature

public Boolean retainAll(Set setOfElementsToRetain)

Parameters

setOfElementsToRetain
Type: Set

Return Value

Type: Boolean

Returns true if the original set changed as a result of the call.

Usage

This method results in the intersection of the two sets. The specified set must be of the same type as the original set that calls the method.

size()

Returns the number of elements in the set (its cardinality).

Signature

public Integer size()

Return Value

Type: Integer

Example

```java
Set<Integer> mySet = new Set<Integer>{1, 2, 3};
List<Integer> myList = new List<Integer>{1, 3};
Boolean result = mySet.retainAll(myList);
System.assertEquals(true, result);

Integer result2 = mySet.size();
System.assertEquals(2, result2);
```

toString()

Returns the string representation of the set.

Signature

public String toString()
Return Value
Type: String

Usage
When used in cyclic references, the output is truncated to prevent infinite recursion. When used with large collections, the output is truncated to avoid exceeding total heap size and maximum CPU time.

- Up to 10 items per collection are included in the output, followed by an ellipsis (…).
- If the same object is included multiple times in a collection, it’s shown in the output only once; subsequent references are shown as (already output).

Site Class
Use the Site Class to manage your sites. Change, reset, validate, and check the expiration of passwords. Create site users, person accounts, and portal users. Get the admin email and ID. Get various URLs, the path prefix, the ID, the template, and the type of the site. Log in to the site.

Namespace
System

Site Methods
The following are methods for Site. All methods are static.

IN THIS SECTION:

changePassword(newPassword, verifyNewPassword, oldPassword)
Changes the password of the current user.
createExternalUser(user, accountId)
Creates a Salesforce Site or Experience Cloud site user for the given account and associates it with the site.
createExternalUser(user, accountId, password)
Creates a Salesforce Site or Experience Cloud site user for the given account and associates it with the site. This method sends an email with the specified password to the user.
createExternalUser(user, accountId, password, sendEmailConfirmation)
Creates a Salesforce Site or Experience Cloud site user and associates it with the given account. This method sends the user an email with the specified password and a new user confirmation email.
createPersonAccountPortalUser(user, ownerId, password)
Creates a person account using the default record type defined on the guest user’s profile, then enables it for the site’s portal.
createPersonAccountPortalUser(user, ownerId, recordTypeId, password)
Creates a person account using the specified recordTypeID, then enables it for the site’s portal.
createPortalUser(user, accountId, password, sendEmailConfirmation)
Creates a portal user for the given account and associates it with the site’s portal.
forgotPassword(username, emailTemplateName)
Resets the user's password and sends an email to the user with the user's new password. You can specify a custom email template or use the default email template. Returns a value indicating whether the password reset was successful.

forgotPassword(username)
Resets the user's password and sends an email to the user with the user's new password. Returns a value indicating whether the password reset was successful.

getAdminEmail()
Returns the email address of the site administrator.

getAdminId()
Returns the user ID of the site administrator.

getAnalyticsTrackingCode()
The tracking code associated with your site. Services such as Google Analytics can use this code to track page request data for your site.

gGetCurrentSiteUrl()
Deprecated. This method was replaced by getBaseUrl() in API version 30.0. Returns the base URL of the current site that references and links should use.

getBaseCustomUrl()
Returns a base URL for the current site that doesn’t use a force.com subdomain. The returned URL uses the same protocol (HTTP or HTTPS) as the current request if at least one non-Force.com custom URL that supports HTTPS exists on the site. The returned value never ends with a / character. If all the custom URLs in this site end in Force.com or this site has no custom URLs, then this returns an empty string. If the current request is not a site request, then this method returns an empty string. This method replaced getCustomWebAddress and includes the custom URL's path prefix.

getBaseInsecureUrl()
Deprecated. Returns a base URL for the current site that uses HTTP instead of HTTPS. The current request’s domain is used. The returned value includes the path prefix and never ends with a / character. If the current request is not a site request, then this method returns an empty string.

gBaseUrl()
Deprecated. Returns a base URL for the current site that uses HTTPS instead of HTTP. The current request’s domain is preferred if it supports HTTPS. Domains that are not Force.com subdomains are preferred over Force.com subdomains. A Force.com subdomain, if associated with the site, is used if no other HTTPS domains exist in the current site. If no HTTPS custom URLs exist in the site, then this method returns an empty string. The returned value includes the path prefix and never ends with a / character. If the current request is not a site request, then this method returns an empty string.

gBaseUrl()
Returns the base URL of the current site that references and links should use. Note that this field may return the referring page's URL instead of the current request's URL. The returned value includes the path prefix and never ends with a / character. If the current request is not a site request, then this field returns an empty string. This field replaces getCurrentSiteUrl.

gCustomWebAddress()
Deprecated. This method was replaced by getBaseCustomUrl() in API version 30.0.

gDomain()
Returns your Salesforce Sites based URL.
getErrorDescription()
Returns the error description for the current page if it’s a designated error page for the site and an error exists; otherwise, returns an empty string.

getErrorMessage()
Returns an error message for the current page if it’s a designated error page for the site and an error exists; otherwise, returns an empty string.

getExperienceId()
Returns the value of the experience ID (expid). This expid value comes from a cookie in the user’s web browser.

getMasterLabel()
Returns the value of the Master Label field for the current site. If the current request is not a site request, then this field returns null.

getName()
Returns the API name of the current site.

getOriginalUrl()
Returns the original URL for this page if it’s a designated error page for the site; otherwise, returns null.

getPasswordPolicyStatement()
Returns the password requirements for a Salesforce Site or Experience Cloud site created with the Customer Service template.

getPathPrefix()
Returns the URL path prefix of the current site or an empty string if none. For example, if the requested site URL is https://myco.my.salesforce-sites.com/partners, then /partners is the path prefix. If the current request is not a site request, then this method returns an empty string. This method replaced getPrefix in API version 30.0.

getPrefix()
Deprecated. This method was replaced by getPathPrefix() in API version 30.0.

getSiteId()
Returns the ID of the current site. If the current request is not a site request, then this field returns null.

getTemplate()
Returns the template name associated with the current site; returns the default template if no template has been designated.

getSiteType()
Returns the API value of the site type field for the current site. This can be Visualforce for a Salesforce site, Siteforce for a Site.com site, ChatterNetwork for an Experience Cloud site, or ChatterNetworkPicasso for an Experience Cloud site. If the current request is not a site request, then this method returns null.

getSiteTypeLabel()
Returns the value of the Site Type field's label for the current site. If the current request is not a site request, then this method returns null.

isLoginEnabled()
Returns true if the current site is associated with an active login-enabled portal; otherwise returns false.

isPasswordExpired()
For authenticated users, returns true if the currently logged-in user's password is expired. For non-authenticated users, returns false.

isRegistrationEnabled()
Returns true if the current site is associated with an active self-registration-enabled Customer Portal; otherwise returns false.

isValidUsername(username)
Returns true if the given username is valid; otherwise, returns false.
login(username, password, startUrl)
Allows users to log in to the current site with the given username and password, then takes them to the startUrl. If startUrl is not a relative path, it defaults to the site’s designated index page.

passwordlessLogin(userId, methods, startUrl)
Logs in a user to a Salesforce Site or Experience Cloud site using an identity verification method, such as email or text, instead of a password. Passwordless login is a convenient, mobile-centric way to welcome users into your site. Let your users log in with something other than their password, like their email address or phone number.

setExperienceId(expIdValue)
Sets the experience ID for the current user. Use this method to populate the value of the experience ID (expid) cookie in the user’s web browser.

setPortalUserAsAuthProvider(user, contactId)
Sets the specified user information within the site’s portal via an authentication provider.

validatePassword(user, password, confirmPassword)
Indicates whether a given password meets the requirements specified by org-wide or profile-based password policies in the current user’s org.

**changePassword(newPassword, verifyNewPassword, oldPassword)**
Changes the password of the current user.

**Signature**

```java
public static System.PageReference changePassword(String newPassword, String verifyNewPassword, String oldPassword)
```

**Parameters**

- `newPassword`
  Type: `String`

- `verifyNewPassword`
  Type: `String`

- `oldPassword`
  Type: `String`

  Optional only if the current user’s password has expired; otherwise, required.

**Return Value**

Type: `System.PageReference`

**Usage**

Calls to this method in API version 30.0 and later can’t commit the transaction automatically. Calls to this method before API version 30.0 commit the transaction, making it impossible to roll back to a save point before the call.

**createExternalUser(user, accountId)**
Creates a Salesforce Site or Experience Cloud site user for the given account and associates it with the site.
Signature

public static Id createExternalUser(SObject user, String accountId)

Parameters

user
  Type: SObject
  Information required to create a user.
  The email address of the user is used to look for matching contacts associated with the specified accountId. If a matching contact is found and is already used by an external user, self-registration isn’t successful. If a matching contact is found but isn’t used by an external user, it is used for the new external user. If there is no matching contact, a new contact is created for the new external user.

accountId
  Type: String
  The ID of the account you want to associate the user with.

Return Value

Type: Id
The ID of the user that this method creates.

Usage

This method throws Site.ExternalUserCreateException when user creation fails.

The nickname field is required for the User sObject when using the createExternalUser method.

Note: This method is only valid when a site is associated with a Customer Portal.

Calls to this method in API version 30.0 and later can’t commit the transaction automatically. Calls to this method before API version 30.0 commit the transaction, making it impossible to roll back to a save point before the call.

createExternalUser(user, accountId, password)

Creates a Salesforce Site or Experience Cloud site user for the given account and associates it with the site. This method sends an email with the specified password to the user.

Signature

public static Id createExternalUser(SObject user, String accountId, String password)

Parameters

user
  Type: SObject
  Information required to create a user.
  The email address of the user is used to look for matching contacts associated with the specified accountId. If a matching contact is found and is already used by an external user, self-registration isn’t successful. If a matching contact is found but isn’t used by an external user, it is used for the new external user. If there is no matching contact, a new contact is created for the new external user.
accountID
Type: String
The ID of the account you want to associate the user with.

password
Type: String
The password of the Salesforce Site or Experience Cloud site user. If not specified, or if set to null or an empty string, this method sends a new password email to the portal user.

Return Value
Type: Id
The ID of the user that this method creates.

Usage
This method throws Site.ExternalUserCreateException when user creation fails.
The nickname field is required for the User sObject when using the createExternalUser method.

Note: This method is only valid when a site is associated with a Customer Portal.

Calls to this method in API version 30.0 and later can’t commit the transaction automatically. Calls to this method before API version 30.0 commit the transaction, making it impossible to roll back to a save point before the call.

createExternalUser(user, accountId, password, sendEmailConfirmation)
Creates a Salesforce Site or Experience Cloud site user and associates it with the given account. This method sends the user an email with the specified password and a new user confirmation email.

Signature
public static Id createExternalUser(SObject user, String accountId, String password, Boolean sendEmailConfirmation)

Parameters
user
Type: SObject
Information required to create a user.
The email address of the user is used to look for matching contacts associated with the specified accountId. If a matching contact is found and is already used by an external user, self-registration isn’t successful. If a matching contact is found but isn’t used by an external user, it is used for the new external user. If there is no matching contact, a new contact is created for the new external user.

accountId
Type: String
The ID of the account you want to associate the user with.

password
Type: String
The password of the Salesforce Site or Experience Cloud site user. If not specified, or if set to `null` or an empty string, this method sends a new password email to the portal user.

`sendEmailConfirmation`  
Type: Boolean  
Determines whether a new user email is sent to the portal user. Set it to `true` to send a new user email to the portal user. The default is `false`, that is, the new user email isn’t sent.

Return Value  
Type: Id  
The ID of the user that this method creates.

Usage  
This method throws `Site.ExternalUserCreateException` when user creation fails.  
The `nickname` field is required for the `User` sObject when using the `createExternalUser` method.

Note: This method is only valid when a site is associated with a Customer Portal.

Calls to this method in API version 30.0 and later can’t commit the transaction automatically. Calls to this method before API version 30.0 commit the transaction, making it impossible to roll back to a save point before the call.

`createPersonAccountPortalUser(user, ownerId, password)`  
Creates a person account using the default record type defined on the guest user’s profile, then enables it for the site’s portal.

Signature  
```java  
public static ID createPersonAccountPortalUser(sObject user, String ownerId, String password)  
```

Parameters  
- `user`  
  Type: `sObject`  
- `ownerId`  
  Type: `String`  
- `password`  
  Type: `String`

Return Value  
Type: ID

Usage  
Calls to this method in API version 30.0 and later can’t commit the transaction automatically. Calls to this method before API version 30.0 commit the transaction, making it impossible to roll back to a save point before the call.
createPersonAccountPortalUser(user, ownerId, recordTypeId, password)

Creates a person account using the specified `recordTypeId`, then enables it for the site's portal.

**Signature**

```
public static ID createPersonAccountPortalUser(sObject user, String ownerId, String recordTypeId, String password)
```

**Parameters**

- **user**
  - Type: `sObject`
- **ownerId**
  - Type: `String`
- **recordTypeId**
  - Type: `String`
- **password**
  - Type: `String`

**Return Value**

Type: `ID`

**Usage**

Calls to this method in API version 30.0 and later can't commit the transaction automatically. Calls to this method before API version 30.0 commit the transaction, making it impossible to roll back to a save point before the call.

**Note:** This method is only valid when a site is associated with a Customer Portal, and when the user license for the default new user profile is a high-volume portal user.

createPortalUser(user, accountId, password, sendEmailConfirmation)

Creates a portal user for the given account and associates it with the site's portal.

**Signature**

```
public static ID createPortalUser(sObject user, String accountId, String password, Boolean sendEmailConfirmation)
```

**Parameters**

- **user**
  - Type: `sObject`
accountId
Type: String

password
Type: String
(Optional) The password of the portal user. If not specified, or if set to null or an empty string, this method sends a new password email to the portal user.

sendEmailConfirmation
Type: Boolean
(Optional) Determines whether a new user email is sent to the portal user. Set it to true to send a new user email to the portal user. The default is false, that is, the new user email isn’t sent.

Return Value
Type: ID

Usage
If you’re using API version 34.0 or later, we recommend using the createExternalUser() methods because they offer better error handling than this method.

The nickname field is required for the user sObject when using the createPortalUser method.

Note: This method is only valid when a site is associated with a Customer Portal.

Calls to this method in API version 30.0 and later can’t commit the transaction automatically. Calls to this method before API version 30.0 commit the transaction, making it impossible to roll back to a save point before the call.

forgotPassword(username, emailTemplateName)
Resets the user’s password and sends an email to the user with the user’s new password. You can specify a custom email template or use the default email template. Returns a value indicating whether the password reset was successful.

Signature
public static Boolean forgotPassword(String username, String emailTemplateName)

Parameters
username
Type: String

emailTemplateName
Type: String
If provided, the method applies the template to the email. Otherwise, the method applies the default system template. If an email template that doesn’t exist is provided, the system logs an exception.

Return Value
Type: Boolean
forgotPassword(username)

Resets the user’s password and sends an email to the user with the user’s new password. Returns a value indicating whether the password reset was successful.

Signature

```
public static Boolean forgotPassword(String username)
```

Parameters

- **username**
  Type: String

Return Value

Type: Boolean

Note: The return value is always true unless it's called outside of a Visualforce page.

Usage

Calls to this method in API version 30.0 and later can’t commit the transaction automatically. Calls to this method before API version 30.0 commit the transaction, making it impossible to roll back to a save point before the call.

Note: Site.forgotPassword cannot be used with the @future method, which enables asynchronous execution.

getAdminEmail()

Returns the email address of the site administrator.

Signature

```
public static String getAdminEmail()
```

Return Value

Type: String

Usage

Calls to this method in API version 30.0 and later can’t commit the transaction automatically. Calls to this method before API version 30.0 commit the transaction, making it impossible to roll back to a save point before the call.

Note: Site.forgotPassword cannot be used with the @future method, which enables asynchronous execution.
**getAdminId()**

Returns the user ID of the site administrator.

Signature

```java
public static ID getAdminId()
```

Return Value

Type: ID

**getAnalyticsTrackingCode()**

The tracking code associated with your site. Services such as Google Analytics can use this code to track page request data for your site.

Signature

```java
public static String getAnalyticsTrackingCode()
```

Return Value

Type: String

**getCurrentSiteUrl()**

Deprecated. This method was replaced by `getBaseUrl()` in API version 30.0. Returns the base URL of the current site that references and links should use.

Note that this may return the referring page's URL instead of the current request's URL. The returned value includes the path prefix and always ends with a `/` character. If the current request is not a site request, then this method returns `null`. If the current request is not a site request, then this method returns `null`. This method was replaced by `getBaseUUrl` in API version 30.0.

Signature

```java
public static String getCurrentSiteUrl()
```

Return Value

Type: String

Usage

Use `getBaseUrl()` instead.

**getBaseCustomUrl()**

Returns a base URL for the current site that doesn’t use a `force.com` subdomain. The returned URL uses the same protocol (HTTP or HTTPS) as the current request if at least one non-Force.com custom URL that supports HTTPS exists on the site. The returned value never ends with a `/` character. If all the custom URLs in this site end in `Force.com` or this site has no custom URLs, then this returns an empty string. If the current request is not a site request, then this method returns an empty string. This method replaced `getCustomWebAddress` and includes the custom URL’s path prefix.
**Signature**

```java
public static String getBaseCustomUrl()
```

**Return Value**

Type: String

**Usage**

This method replaces `getCustomWebAddress()` and includes the custom URL's path prefix.

**getBaseInsecureUrl()**

Deprecated. Returns a base URL for the current site that uses HTTP instead of HTTPS. The current request's domain is used. The returned value includes the path prefix and never ends with a `/` character. If the current request is not a site request, then this method returns an empty string.

**Signature**

```java
public static String getBaseInsecureUrl()
```

**Return Value**

Type: String

**getBaseRequestUrl()**

Returns the base URL of the current site for the requested URL. This isn't influenced by the referring page's URL. The returned URL uses the same protocol (HTTP or HTTPS) as the current request. The returned value includes the path prefix and never ends with a `/` character. If the current request is not a site request, then this method returns an empty string.

**Signature**

```java
public static String getBaseRequestUrl()
```

**Return Value**

Type: String

**getBaseSecureUrl()**

Returns a base URL for the current site that uses HTTPS instead of HTTP. The current request's domain is preferred if it supports HTTPS. Domains that are not Force.com subdomains are preferred over Force.com subdomains. A Force.com subdomain, if associated with the site, is used if no other HTTPS domains exist in the current site. If no HTTPS custom URLs exist in the site, then this method returns an empty string. The returned value includes the path prefix and never ends with a `/` character. If the current request is not a site request, then this method returns an empty string.

**Signature**

```java
public static String getBaseSecureUrl()
```
Return Value
Type: String

**getBaseUrl()**
Returns the base URL of the current site that references and links should use. Note that this field may return the referring page's URL instead of the current request's URL. The returned value includes the path prefix and never ends with a / character. If the current request is not a site request, then this field returns an empty string. This field replaces getCurrentSiteUrl.

Signature

```java
public static String getBaseUrl()
```

Return Value
Type: String

Usage
This method replaces `getCurrentSiteUrl()`.

**getCustomWebAddress()**
Deprecated. This method was replaced by `getBaseCustomUrl()` in API version 30.0.
Returns the request's custom URL if it doesn't end in Lightning Platform or returns the site's primary custom URL. If neither exist, then this returns null. Note that the URL's path is always the root, even if the request's custom URL has a path prefix. If the current request is not a site request, then this method returns null. The returned value always ends with a / character.

Signature

```java
public static String getCustomWebAddress()
```

Return Value
Type: String

Usage
Use `getBaseCustomUrl()` instead.

**getDomain()**
Returns your Salesforce Sites based URL.

Signature

```java
public static String getDomain()
```
Return Value
Type: String

**getErrorDescription()**
Returns the error description for the current page if it’s a designated error page for the site and an error exists; otherwise, returns an empty string.

**Signature**

```java
public static String getErrorDescription()
```

Return Value
Type: String

**getErrorMessage()**
Returns an error message for the current page if it’s a designated error page for the site and an error exists; otherwise, returns an empty string.

**Signature**

```java
public static String getErrorMessage()
```

Return Value
Type: String

**getExperienceId()**
Returns the value of the experience ID (expid). This expid value comes from a cookie in the user’s web browser.

**Signature**

```java
public static String getExperienceId()
```

Return Value
Type: String

**Usage**
Use the `getExperienceId` and `setExperienceId` methods to implement dynamic login experiences. You can set the experience ID with `setExperienceId` or by extending the following endpoints with `expid_value`:

- `community-url/services/oauth2/authorize/expid_value`
- `community-url/idp/endpoint/HttpPost/expid_value`
- `community-url/idp/endpoint/HttpRedirect/expid_value`
- `community-url_login_page/expid={value}`
The cookie is set when the browser loads the URLs with the expid values.

**getMasterLabel()**

Returns the value of the Master Label field for the current site. If the current request is not a site request, then this field returns `null`.

**Signature**

```java
public static String getMasterLabel()
```

**Return Value**

Type: `String`

**getName()**

Returns the API name of the current site.

**Signature**

```java
public static String getName()
```

**Return Value**

Type: `String`

**getOriginalUrl()**

Returns the original URL for this page if it’s a designated error page for the site; otherwise, returns `null`.

**Signature**

```java
public static String getOriginalUrl()
```

**Return Value**

Type: `String`

**getPasswordPolicyStatement()**

Returns the password requirements for a Salesforce Site or Experience Cloud site created with the Customer Service template.

**Signature**

```java
public static String getPasswordPolicyStatement()
```
Return Value
Type: String

getPathPrefix()
Returns the URL path prefix of the current site or an empty string if none. For example, if the requested site URL is https://myco.my.salesforce-sites.com/partners, then /partners is the path prefix. If the current request is not a site request, then this method returns an empty string. This method replaced getPrefix in API version 30.0.

Signature
public static String getPathPrefix()

Return Value
Type: String

getPrefix()
Deprecated. This method was replaced by getPathPrefix() in API version 30.0.
Returns the URL path prefix of the current site. For example, if your site URL is MyDomainName.my.salesforce-sites.com/partners, /partners is the path prefix. Returns null if the prefix isn’t defined. If the current request is not a site request, then this method returns a null.

Signature
public static String getPrefix()

Return Value
Type: String

getSiteId()
Returns the ID of the current site. If the current request is not a site request, then this field returns null.

Signature
public static String getSiteId()

Return Value
Type: Id

getTemplate()
Returns the template name associated with the current site; returns the default template if no template has been designated.
**Signature**

```java
public static System.PageReference getTemplate()
```

**Return Value**
Type: `System.PageReference`

---

**getSiteType()**

Returns the API value of the site type field for the current site. This can be Visualforce for a Salesforce site, Siteforce for a Site.com site, ChatterNetwork for an Experience Cloud site, or ChatterNetworkPicasso for an Experience Cloud site. If the current request is not a site request, then this method returns `null`.

**Signature**

```java
public static String getSiteType()
```

**Return Value**
Type: `String`

---

**getSiteTypeLabel()**

Returns the value of the Site Type field's label for the current site. If the current request is not a site request, then this method returns `null`.

**Signature**

```java
public static String getSiteTypeLabel()
```

**Return Value**
Type: `String`

---

**isLoginEnabled()**

Returns `true` if the current site is associated with an active login-enabled portal; otherwise returns `false`.

**Signature**

```java
public static Boolean isLoginEnabled()
```

**Return Value**
Type: `Boolean`

---

**isPasswordExpired()**

For authenticated users, returns `true` if the currently logged-in user's password is expired. For non-authenticated users, returns `false`.

---

3366
Signature
public static Boolean isPasswordExpired()

Return Value
Type: Boolean

isRegistrationEnabled()
Returns true if the current site is associated with an active self-registration-enabled Customer Portal; otherwise returns false.

Signature
public static Boolean isRegistrationEnabled()

Return Value
Type: Boolean

isValidUsername(username)
Returns true if the given username is valid; otherwise, returns false.

Signature
public static Boolean isValidUsername(String username)

Parameters
username
Type: String
The username to test for validity.

Return Value
Type: Boolean

login(username, password, startUrl)
Allows users to log in to the current site with the given username and password, then takes them to the startUrl. If startUrl is not a relative path, it defaults to the site's designated index page.

Signature
public static System.PageReference login(String username, String password, String startUrl)
Parameters

- **username**
  Type: String

- **password**
  Type: String

- **startUrl**
  Type: String

Return Value

Type: System.PageReference

Usage

All DML statements before the call to Site.login get committed. It's not possible to roll back to a save point that was created before a call to Site.login.

暄Note: Do not include http:// or https:// in the startURL.

passwordlessLogin(userId, methods, startUrl)

Logs in a user to a Salesforce Site or Experience Cloud site using an identity verification method, such as email or text, instead of a password. Passwordless login is a convenient, mobile-centric way to welcome users into your site. Let your users log in with something other than their password, like their email address or phone number.

Signature

```
public static System.PageReference passwordlessLogin(Id userId, List<Auth.VerificationMethod> methods, String startUrl)
```

Parameters

- **userId**
  Type: Id
  ID of the user to log in.

- **methods**
  Type: List<Auth.VerificationMethod>
  List of identity verification methods available to the user for passwordless login.

- **startUrl**
  Type: String
  Path to the page that users see after they log in.

Return Value

Type: System.PageReference
Usage
Include this method in the Apex controller of a custom login page implementation.

PasswordlessLogin Example
This simple code example of an Apex controller contains the passwordlessLogin method. The PageReference returned by passwordlessLogin redirects the user to the Salesforce Verify page. When the user enters the correct code, the user is redirected to the site page specified by the start URL.

```java
global with sharing class MFILoginController
{
    // Input variables
    global String input {get; set;}
    public String startURL {get; set;}
    public List<Auth.VerificationMethod> methods;
    public String error;

    global MFILoginController()
    {
        // Add verification methods in priority order
        methods = new List<Auth.VerificationMethod>();
        methods.add(Auth.VerificationMethod.SMS);
        methods.add(Auth.VerificationMethod.EMAIL);
        methods.add(Auth.VerificationMethod.U2F);
        methods.add(Auth.VerificationMethod.SALESFORCE_AUTHENTICATOR);
        methods.add(Auth.VerificationMethod.TOTP);
    }

    global PageReference login() {
        List<User> users = null;
        // Empty input
        if(input == null || input == '')
        {
            error = 'Enter Username';
            return null;
        }
        users = [select name, id, email from User where username=:input];
        if(users == null || users.isEmpty())
        {
            error = 'Can\'t find a user';
            return null;
        }
        if (startURL == null) startURL = '/';
        return Site.passwordlessLogin(users[0].id, methods, startURL);
    }
}
```
setExperienceId(expIdValue)

Sets the experience ID for the current user. Use this method to populate the value of the experience ID (expid) cookie in the user’s web browser.

**Signature**

```java
public static void setExperienceId(String expIdValue)
```

**Parameters**

`expIdValue`

Type: String

A value that indicates the user's login experience.

The value must contain alphanumeric characters only, up to 30 characters.

**Usage**

Use `setExperienceId` when you’re implementing dynamic login experiences. A login experience refers to a login page plus any secondary pages associated with the login page (such as multi-factor authentication (MFA) or a login flow). You define different login experiences depending on who users are or where they’re logging in from. For example, you can require a different registration process based on the user’s location. In this case, `expIdValue` includes a state or country code. When the user logs in, the URL contains the experience ID parameter, `{expid}`. The `{expid}` parameter is replaced by the value stored in `expIdValue`, such as `.jp`. Then the user is redirected to the Japanese login experience.

**Example**

```java
String expid = ApexPages.currentPage().getParameters().get('expid');
if (expId != null) {
    Site.setExperienceId(expId);
}
```

setPortalUserAsAuthProvider(user, contactId)

Sets the specified user information within the site’s portal via an authentication provider.

**Signature**

```java
public static Void setPortalUserAsAuthProvider(sObject user, String contactId)
```

**Parameters**

`user`

Type: sObject

`contactId`

Type: String

**Return Value**

Type: Void
Usage

- This method is only valid when a site is associated with a Customer Portal.
- Calls to this method in API version 30.0 and later can’t commit the transaction automatically. Calls to this method before API version 30.0 commit the transaction, making it impossible to roll back to a save point before the call.
- For more information on an authentication provider, see RegistrationHandler.

validatePassword(user, password, confirmPassword)

Indicates whether a given password meets the requirements specified by org-wide or profile-based password policies in the current user’s org.

Signature

public static void validatePassword(SObject user, String password, String confirmPassword)

Parameters

user
Type: SObject
The user attempting to create a password during self-registration for a Salesforce Site or Experience Cloud site.

password
Type: String
The password entered by the user.

confirmPassword
Type: String
The password reentered by the user to confirm the password.

Return Value

Type: void

Usage

If validation fails when the method is run in a Lightning controller, this method throws an Apex exception describing the failed validation.
If validation fails when the method is run in a Visualforce controller, the method provides Visualforce error messages.

SObject Class

Contains methods for the sObject data type.

Namespace

System
Usage

SObject methods are all instance methods: they are called by and operate on an sObject instance such as an account or contact. The following are the instance methods for sObjects.

For more information on sObjects, see Working with sObjects.

SObject Methods

The following are methods for SObject. All are instance methods.

IN THIS SECTION:

addError(errorMsg)
Marks a trigger record with a custom error message and prevents any DML operation from occurring.

addError(errorMsg, escape)
Marks a trigger record with a custom error message, specifies if the error message should be escaped, and prevents any DML operation from occurring.

addError(exceptionError)
Marks a trigger record with a custom error message and prevents any DML operation from occurring.

addError(exceptionError, escape)
Marks a trigger record with a custom exception error message, specifies whether or not the exception error message should be escaped, and prevents any DML operation from occurring.

addError(fieldName, errorMsg)
Dynamically add errors to fields of an SObject associated with the specified field name.

addError(fieldToken, errorMsg)
Dynamically add errors to an SObject instance associated with the specified field.

addError(fieldName, errorMsg, escape)
Dynamically add errors to fields of an SObject associated with the specified field name.

addError(fieldToken, errorMsg, escape)
Dynamically add errors to an SObject instance associated with the specified field.

clear()
Clears all field values

clone(preserveId, isDeepClone, preserveReadOnlyTimestamps, preserveAutonumber)
Creates a copy of the SObject record.

get(fieldName)
Returns the value for the field specified by fieldName, such as AccountNumber.
get(field)
Returns the value for the field specified by the field token Schema.sObjectField such as, Schema.Account.AccountNumber.

getCloneSourceId()
Returns the ID of the entity from which an object was cloned. You can use it for objects cloned through the Salesforce user interface. You can also use it for objects created using the System.SObject.clone(preserveId, isDeepClone, preserveReadOnlyTimestamps, preserveAutonumber) method, provided that the preserveId parameter wasn't used or was set to false. The getCloneSourceId() method can only be used within the transaction where the entity is cloned, as clone information doesn't persist in subsequent transactions.

getErrors()
Returns a list of Database.Error objects for an SObject instance. If the SObject has no errors, an empty list is returned.

getOptions()
Returns the database.DMLOptions object for the SObject.

getPopulatedFieldsAsMap()
Returns a map of populated field names and their corresponding values. The map contains only the fields that have been populated in memory for the SObject instance.

getSObject(fieldName)
Returns the value for the specified field. This method is primarily used with dynamic DML to access values for external IDs.

getSObject(field)
Returns the value for the field specified by the field token Schema.sObjectField such as, Schema.MyObj.MyExternalId. This method is primarily used with dynamic DML to access values for external IDs.

getSObjects(fieldName)
Returns the values for the specified field. This method is primarily used with dynamic DML to access values for associated objects, such as child relationships.

getSObjects(field)
Returns the value for the field specified by the field token Schema.fieldName, such as, Schema.Account.Contact. This method is primarily used with dynamic DML to access values for associated objects, such as child relationships.

getSObjectType()
Returns the token for this SObject. This method is primarily used with describe information.

getQuickActionName()
Retrieves the name of a quick action associated with this SObject. Typically used in triggers.

hasErrors()
Returns true if an SObject instance has associated errors. The error message can be associated to the SObject instance by using SObject.addError(), validation rules, or by other means.

isClone()
Returns true if an entity is cloned from something, even if the entity hasn't been saved. The method can only be used within the transaction where the entity is cloned, as clone information doesn't persist in subsequent transactions.

isSet(fieldName)
Returns information about the queried SObject field. Returns true if the SObject field is populated, either by direct assignment or by inclusion in a SOQL query. Returns false if the SObject field is not set. If an invalid field is specified, an SObjectException is thrown.
isSet(field)
Returns information about the queried sObject field. Returns true if the sObject field is populated, either by direct assignment or by inclusion in a SOQL query. Returns false if the sObject field is not set. If an invalid field is specified, an SObjectException is thrown.

put(fieldName, value)
Sets the value for the specified field and returns the previous value for the field.

put(field, value)
Sets the value for the field specified by the field token Schema.sObjectField, such as, Schema.Account.AccountNumber and returns the previous value for the field.

putSObject(fieldName, value)
Sets the value for the specified field. This method is primarily used with dynamic DML for setting external IDs. The method returns the previous value of the field.

putSObject(field, value)
Sets the value for the field specified by the token Schema.SObjectType. This method is primarily used with dynamic DML for setting external IDs. The method returns the previous value of the field.

recalculateFormulas()
Deprecated as of API version 57.0. Use the recalculateFormulas() method in the System.Formula class instead.

setOptions(DMLOptions)
Sets the DMLOptions object for the SObject.

addError(errorMsg)
Marks a trigger record with a custom error message and prevents any DML operation from occurring.

Signature
public Void addError(String errorMsg)

Parameters
errorMsg
  Type: String
  The error message to mark the record with.

Return Value
Type: Void

Usage
When used on Trigger.new in before insert and before update triggers, and on Trigger.old in before delete triggers, the error message is displayed in the application interface.
See Triggers and Trigger Exceptions.
**Note:** This method escapes any HTML markup in the specified error message. The escaped characters are: \n, <, >, &, "", \u2028, \u2029, and \u00a9. As a result, HTML markup is not rendered; instead, it is displayed as text in the Salesforce user interface.

When used in Visualforce controllers, the generated message is added to the collection of errors for the page. For more information, see Validation Rules and Standard Controllers in the Visualforce Developer’s Guide.

**Example**

```java
Trigger.new[0].addError('bad');
```

### addError(errorMsg, escape)

Marks a trigger record with a custom error message, specifies if the error message should be escaped, and prevents any DML operation from occurring.

**Signature**

```java
public Void addError(String errorMsg, Boolean escape)
```

**Parameters**

- **errorMsg**
  - Type: `String`
  - The error message to mark the record with.

- **escape**
  - Type: `Boolean`
  - Indicates whether any HTML markup in the custom error message should be escaped (`true`) or not (`false`). This parameter is ignored in both Lightning Experience and the Salesforce mobile app, and the HTML is always escaped. The escape parameter only applies in Salesforce Classic.

**Return Value**

Type: Void

**Usage**

The escaped characters are: \n, <, >, &, "", \u2028, \u2029, and \u00a9. As a result, HTML markup is not rendered; instead, it is displayed as text in the Salesforce user interface.

**Warning:** Be cautious if you specify `false` for the `escape` argument. Unescaped strings displayed in the Salesforce user interface can represent a vulnerability in the system because these strings might contain harmful code. If you want to include HTML markup in the error message, call this method with a `false escape` argument. Make sure that you escape any dynamic content, such as input field values. Otherwise, specify `true` for the `escape` argument or call `addError(String errorMsg)` instead.

**Example**

```java
Trigger.new[0].addError('Fix & resubmit', false);
```
**addError(exceptionError)**
Marks a trigger record with a custom error message and prevents any DML operation from occurring.

**Signature**

```java
public Void addError(Exception exceptionError)
```

**Parameters**

`exceptionError`
Type: `System.Exception`
An Exception object or a custom exception object that contains the error message to mark the record with.

**Return Value**
Type: Void

**Usage**
When used on `Trigger.new` in `before insert` and `before update` triggers, and on `Trigger.old` in `before delete` triggers, the error message is displayed in the application interface.

See [Triggers and Trigger Exceptions](#).

**Note:** This method escapes any HTML markup in the specified error message. The escaped characters are: \n, <, >, &", \", \u2028, \u2029, and \u00a9. As a result, HTML markup is not rendered; instead, it is displayed as text in the Salesforce user interface.

When used in Visualforce controllers, the generated message is added to the collection of errors for the page. For more information, see [Validation Rules and Standard Controllers](#) in the Visualforce Developer's Guide.

**Example**

```java
public class MyException extends Exception {
}
Trigger.new[0].addError(new MyException('Invalid Id'));
```

**addError(exceptionError, escape)**
Marks a trigger record with a custom exception error message, specifies whether or not the exception error message should be escaped, and prevents any DML operation from occurring.

**Signature**

```java
public Void addError(Exception exceptionError, Boolean escape)
```

**Parameters**

`exceptionError`
Type: `System.Exception`
An Exception object or a custom exception object that contains the error message to mark the record with.
escape
Type: Boolean
Indicates whether any HTML markup in the custom error message should be escaped (true) or not (false). This parameter is ignored in both Lightning Experience and the Salesforce mobile app, and the HTML is always escaped. The escape parameter only applies in Salesforce Classic.

Return Value
Type: Void

Usage
The escaped characters are: \n, <, >, &. As a result, HTML markup is not rendered; instead, it is displayed as text in the Salesforce user interface.

⚠️ Warning: Be cautious if you specify false for the escape argument. Unescaped strings displayed in the Salesforce user interface can represent a vulnerability in the system because these strings might contain harmful code. If you want to include HTML markup in the error message, call this method with a false escape argument. Make sure that you escape any dynamic content, such as input field values. Otherwise, specify true for the escape argument or call addError(Exception e) instead.

Example
```java
public class MyException extends Exception {}
Trigger.new[0].addError(new myException('Invalid Id & other issues', false));
```

addError(errorMsg)
Places the specified error message on a trigger record field in the Salesforce user interface and prevents any DML operation from occurring.

Signature
```java
public Void addError(String errorMsg)
```

Parameters
```java
errorMsg
Type: String
```

Return Value
Type: Void

Usage
Note:
- When used on Trigger.new in before insert and before update triggers, and on Trigger.old in before delete triggers, the error appears in the application interface.
• When used in Visualforce controllers, if there is an `inputField` component bound to field, the message is attached to the component. For more information, see Validation Rules and Standard Controllers in the Visualforce Developer's Guide.

• This method is highly specialized because the field identifier is not actually the invoking object—the sObject record is the invoker. The field is simply used to identify the field that should be used to display the error.

See Triggers and Trigger Exceptions.

Note: This method escapes any HTML markup in the specified error message. The escaped characters are: \n, <, >, &, "", \, \u2028, \u2029, and \u00a9. As a result, HTML markup is not rendered; instead, it is displayed as text in the Salesforce user interface.

Example

```java
Trigger.new[0].myField__c.addError('bad');
```

`addError(errorMsg, escape)`

Places the specified error message, which can be escaped or unescaped, on a trigger record field in the Salesforce user interface, and prevents any DML operation from occurring.

**Signature**

```java
public Void addError(String errorMsg, Boolean escape)
```

**Parameters**

- `errorMsg`
  Type: `String`
  The error message to mark the record with.

- `escape`
  Type: `Boolean`
  Indicates whether any HTML markup in the custom error message should be escaped (true) or not (false). This parameter is ignored in both Lightning Experience and the Salesforce mobile app, and the HTML is always escaped. The escape parameter only applies in Salesforce Classic.

**Return Value**

Type:

**Usage**

The escaped characters are: \n, <, >, &, "", \, \u2028, \u2029, and \u00a9. As a result, HTML markup is not rendered; instead, it is displayed as text in the Salesforce user interface.

⚠️ **Warning**: Be cautious if you specify `false` for the `escape` argument. Unescaped strings displayed in the Salesforce user interface can represent a vulnerability in the system because these strings might contain harmful code. If you want to include HTML markup in the error message, call this method with a `false` `escape` argument. Make sure that you escape any dynamic content, such as input field values. Otherwise, specify `true` for the `escape` argument or call `field.addError(String errorMsg)` instead.
Example

```
Trigger.new[0].myField__c.addError('Fix & resubmit', false);
```

addError(fieldName, errorMsg)
Dynamically add errors to fields of an SObject associated with the specified field name.

Signature
```
public void addError(String fieldName, String errorMsg)
```

Parameters

fieldName
Type: String
The field name of the SObject.

errorMsg
Type: String
The error message to be added. HTML special characters in the error message string are always escaped.

Return Value
Type: void

Usage
If the field name is an empty string or null, the error is associated with the SObject and not with a specific field.

Example

```
// Add an error to an SObject field using the addError() method.
Account acct = new Account(name = 'TestAccount');
acct.addError('name', 'error in name field');
// Use the hasErrors() method to verify that the error is added, and then the getErrors() method to validate the error.
System.Assert(acct.hasErrors());
List<Database.Error> errors = acct.getErrors();
System.AssertEquals(1, errors.size());
```

addError(fieldToken, errorMsg)
Dynamically add errors to an SObject instance associated with the specified field.

Signature
```
public void addError(Schema.SObjectField fieldToken, String errorMsg)
```

Usage
If the field name is an empty string or null, the error is associated with the SObject and not with a specific field.
Parameters

`fieldToken`
- **Type:** Schema.SObjectField
- The field of the SObject instance.

`errorMsg`
- **Type:** String
- The error message to be added. HTML special characters in the error message string are always escaped.

Return Value

**Type:** void

Usage

Use this method to add errors to the specified field token of a standard or custom object. If `fieldToken` is null, the error is associated with the SObject and not with a specific field.

Example

```java
// Add an error to a field of an SObject instance using the addError() method.
Account acct = new Account(name = 'TestAccount');
Schema.DescribeFieldResult nameDesc = Account.name.getDescribe();
Schema.sObjectField nameField = nameDesc.getSObjectField();
acct.addError(nameField, 'error is name field');
// Use the hasErrors() method to verify that the error is added, and then the getErrors() method to validate the error.
System.Assert(acct.hasErrors());
List<Database.Error> errors = acct.getErrors();
System.AssertEquals(1, errors.size());
```

`addError(fieldName, errorMsg, escape)`

Dynamically add errors to fields of an SObject associated with the specified field name.

**Signature**

```java
public void addError(String fieldName, String errorMsg, Boolean escape)
```

**Parameters**

`fieldName`
- **Type:** String
  - The field name of the SObject.

`errorMsg`
- **Type:** String
  - The error message to be added.

`escape`
- **Type:** Boolean
Indicates whether any HTML markup in the custom error message should be escaped (true) or not (false). This parameter is ignored in both Lightning Experience and the Salesforce mobile app, and the HTML is always escaped. The escape parameter only applies in Salesforce Classic.

Return Value
Type: void

Usage
If the field name is an empty string or null, the error is associated with the SObject and not with a specific field.

The escaped characters are: \n, <, >, &", \u2028, \u2029, and \u00a9. As a result, HTML markup is not rendered; instead, it is displayed as text in the Salesforce user interface.

⚠️ Warning:
- The escape parameter cannot be disabled in Lightning Experience and in the Salesforce mobile app, and will be ignored.
- Be cautious if you specify false for the escape argument. Unescaped strings displayed in the Salesforce user interface can represent a vulnerability in the system because these strings might contain harmful code. If you want to include HTML markup in the error message, call this method with a false escape argument. Make sure that you escape any dynamic content, such as input field values. Otherwise, specify true for the escape argument or call addError(String fieldName, String errorMsg) instead.

Example

```java
// Add an error to an SObject field using the addError() method.
Account acct = new Account(name = 'TestAccount');
acct.addError('name', 'error in name field', false);
// Use the hasErrors() method to verify that the error is added, and then the getErrors() method to validate the error.
System.Assert(acct.hasErrors());
List<Database.Error> errors = acct.getErrors();
System.AssertEquals(1, errors.size());
```

addError(fieldToken, errorMsg, escape)
Dynamically add errors to an SObject instance associated with the specified field.

Signature
```
public void addError(Schema.SObjectField fieldToken, String errorMsg, Boolean escape)
```

Parameters

- **fieldToken**
  Type: Schema.SObjectField
  The field of the SObject instance.

- **errorMsg**
  Type: String
The error message to be added.

escape
Type: Boolean
Indicates whether any HTML markup in the custom error message should be escaped (true) or not (false). This parameter is ignored in both Lightning Experience and the Salesforce mobile app, and the HTML is always escaped. The escape parameter only applies in Salesforce Classic.

Return Value
Type: void

Usage
Use this method to add errors to the specified field token of a standard or custom object. If fieldToken is null, the error is associated with the SObject and not with a specific field.

The escaped characters are: \n, <, >, &, \, \u2028, \u2029, and \u00a9. As a result, HTML markup is not rendered; instead, it is displayed as text in the Salesforce user interface.

Warning:
- The escape parameter cannot be disabled in Lightning Experience and in the Salesforce mobile app, and will be ignored.
- Be cautious if you specify false for the escape argument. Unescaped strings displayed in the Salesforce user interface can represent a vulnerability in the system because these strings might contain harmful code. If you want to include HTML markup in the error message, call this method with a false escape argument. Make sure that you escape any dynamic content, such as input field values. Otherwise, specify true for the escape argument or call addError(Schema.SObjectField fieldToken, String errorMsg) instead.

Example
```
// Add an error to a field of an SObject instance using the addError() method.
Account acct = new Account(name = 'TestAccount');
Schema.DescribeFieldResult nameDesc = Account.name.getDescribe();
Schema.sObjectField nameField = nameDesc.getSObjectField();
acct.addError(nameField, 'error is name field', false);
// Use the hasErrors() method to verify that the error is added, and then the getErrors() method to validate the error.
System.Assert(acct.hasErrors());
List<Database.Error> errors = acct.getErrors();
System.AssertEquals(1, errors.size());
```

clear()
Clears all field values

Signature
```
public Void clear()
```
Return Value
Type: Void

Example

```java
Account acc = new account(Name = 'Acme');
acc.clear();
Account expected = new Account();
system.assertEquals(expected, acc);
```

colors.map Ön, isDeepClone, preserveReadonlyTimestamps, preserveAutonumber)

Creates a copy of the SObject record.

Signature

```java
public SObject clone(Boolean preserveId, Boolean isDeepClone, Boolean preserveReadonlyTimestamps, Boolean preserveAutonumber)
```

Parameters

**preserveId**
Type: Boolean

(Optional) Determines whether the ID of the original object is preserved or cleared in the duplicate. If set to true, the ID is copied to the duplicate. The default is false, that is, the ID is cleared.

**isDeepClone**
Type: Boolean

(Optional) Determines whether the method creates a full copy of the SObject field or just a reference:

- If set to true, the method creates a full copy of the SObject. All fields on the SObject are duplicated in memory, including relationship fields. Consequently, if you change a field on the cloned SObject, the original SObject isn’t affected.
- If set to false, the method performs a shallow copy of the SObject fields. All copied relationship fields reference the original SObjects. Consequently, if you change a relationship field on the cloned SObject, the corresponding field on the original SObject is also affected, and vice versa. The default is false.

**preserveReadonlyTimestamps**
Type: Boolean

(Optional) Determines whether the read-only timestamp fields are preserved or cleared in the duplicate. If set to true, the read-only fields CreatedById, CreatedDate, LastModifiedById, and LastModifiedDate are copied to the duplicate. The default is false, that is, the values are cleared.

Tip: Audit field values won’t be persisted to the database via DML on the cloned SObject instance.

**preserveAutonumber**
Type: Boolean

(Optional) Determines whether auto number fields of the original object are preserved or cleared in the duplicate. If set to true, auto number fields are copied to the cloned object. The default is false, that is, auto number fields are cleared.
Return Value
Type: SObject (of the same type)

Usage
Note: For Apex saved using Salesforce API version 22.0 or earlier, the default value for the preserveId argument is true, that is, the ID is preserved.

Example
Account acc = new account(Name = 'Acme', Description = 'Acme Account');
Account clonedAcc = acc.clone(false, false, false, false);
System.assertEquals(acc, clonedAcc);

get(fieldName)
Returns the value for the field specified by fieldName, such as AccountNumber.

Signature
public Object get(String fieldName)

Parameters
fieldName
Type: String

Return Value
Type: Object

Usage
For more information, see Dynamic SOQL.

Example
Account acc = new account(Name = 'Acme', Description = 'Acme Account');
String description = (String)acc.get('Description');
System.assertEquals('Acme Account', description);

Versioned Behavior Changes
In API version 34.0 and later, you must include the namespace name to retrieve a field from a field Map using this method. For example, to get the account__c field in the MyNamespace namespace from a fields field Map, use:
fields.get('MyNamespace__account__c').
get(field)
Returns the value for the field specified by the field token `Schema.sObjectField` such as, `Schema.Account.AccountNumber`.

Signature
```
public Object get(Schema.sObjectField field)
```

Parameters
- `field`
  Type: `Schema.SObjectField`

Return Value
Type: `Object`

Usage
For more information, see Dynamic SOQL.

Note: Field tokens aren’t available for person accounts. If you access `Schema.Account.fieldname`, you get an exception error. Instead, specify the field name as a string.

Example
```
Account acc = new account(Name = 'Acme', Description = 'Acme Account');
String description = (String)acc.get(Schema.Account.Description);
System.assertEquals('Acme Account', description);
```

getCloneSourceId()
Returns the ID of the entity from which an object was cloned. You can use it for objects cloned through the Salesforce user interface. You can also use it for objects created using the `System.SObject.clone(preserveId, isDeepClone, preserveReadOnlyTimestamps, preserveAutonumber)` method, provided that the `preserveId` parameter wasn’t used or was set to `false`. The `getCloneSourceId()` method can only be used within the transaction where the entity is cloned, as clone information doesn’t persist in subsequent transactions.

Signature
```
public Id getCloneSourceId()
```

Return Value
Type: `Id`

Usage
If A is cloned to B, B is cloned to C, and C is cloned to D, then B, C, and D all point back to A as their clone source.

Example

```java
Account acc0 = new Account(Name = 'Acme');
insert acc0;
Account acc1 = acc0.clone();
Account acc2 = acc1.clone();
Account acc3 = acc2.clone();
Account acc4 = acc3.clone();
System.assert(acc0.Id != null);
System.assertEquals(acc0.Id, acc1.getCloneSourceId());
System.assertEquals(acc0.Id, acc2.getCloneSourceId());
System.assertEquals(acc0.Id, acc3.getCloneSourceId());
System.assertEquals(acc0.Id, acc4.getCloneSourceId());
System.assertEquals(null, acc0.getCloneSourceId());
```

getErrors()

Returns a list of `Database.Error` objects for an SObject instance. If the SObject has no errors, an empty list is returned.

**Signature**

```java
public List<Database.Error> getErrors()
```

**Return Value**

Type: `List<Database.Error>`

getOptions()

Returns the `Database.DMLOptions` object for the SObject.

**Signature**

```java
public Database.DMLOptions getOptions()
```

**Return Value**

Type: `Database.DMLOptions`

Example

```java
Database.DMLOptions dmo = new Database.dmlOptions();
dmo.assignmentRuleHeader.useDefaultRule = true;

Account acc = new Account(Name = 'Acme');
acc.setOptions(dmo);
Database.DMLOptions accDmo = acc.getOptions();
```

getPopulatedFieldsAsMap()

Returns a map of populated field names and their corresponding values. The map contains only the fields that have been populated in memory for the SObject instance.
signature

public Map<String, Object> getPopulatedFieldsAsMap()

return value

Type: Map<String, Object>

A map of field names and their corresponding values.

usage

The returned map contains only the fields that have been populated in memory for the SObject instance, which makes it easy to iterate over those fields. A field is populated in memory in the following cases.

• The field has been queried by a SOQL statement.
• The field has been explicitly set before the call to the getPopulatedFieldsAsMap() method.

Fields on related objects that are queried or set are also returned in the map.

The following example iterates over the map returned by the getPopulatedFieldsAsMap() method after a SOQL query.

```java
Account a = new Account();
a.name = 'TestMapAccount1';
insert a;
a = [select Id, Name from Account where id=:a.Id];
Map<String, Object> fieldsToValue = a.getPopulatedFieldsAsMap();

for (String fieldName : fieldsToValue.keySet()) {
    System.debug('field name is ' + fieldName + ', value is ' + fieldsToValue.get(fieldName));
}

// Example debug statement output:
// DEBUG|field name is Id, value is 001R0000003EPPkIAO
// DEBUG|field name is Name, value is TestMapAccount1
```

This example iterates over the map returned by the getPopulatedFieldsAsMap() method after fields on the SObject are explicitly set.

```java
Account a = new Account();
a.name = 'TestMapAccount2';
a.phone = '123-4567';
insert a;
Map<String, Object> fieldsToValue = a.getPopulatedFieldsAsMap();

for (String fieldName : fieldsToValue.keySet()) {
    System.debug('field name is ' + fieldName + ', value is ' + fieldsToValue.get(fieldName));
}

// Example debug statement output:
// DEBUG|field name is Name, value is TestMapAccount2
// DEBUG|field name is Phone, value is 123-4567
// DEBUG|field name is Id, value is 001R0000003EPPpIAO
```
The following example shows how to use the `getPopulatedFieldsAsMap()` method with related objects.

```java
Account a = new Account();
a.name='TestMapAccount3';
insert a;

Contact c = new Contact();
c.firstname='TestContactFirstName';
c.lastName = 'TestContactLastName';
c.accountid = a.id;
insert c;

c = [SELECT id, Contact.Firstname, Contact.Account.Name FROM Contact
    where id=:c.id limit 1];
Map<String, Object> fieldsToValue = c.getPopulatedFieldsAsMap();

// To get the fields on Account, get the Account object
// and call getMapPopulatedFieldsAsMap() on that object.

a = (Account)fieldsToValue.get('Account');
fieldsToValue = a.getPopulatedFieldsAsMap();

for (String fieldName : fieldsToValue.keySet()) {
    System.debug('field name is ' + fieldName + ', value is ' + fieldsToValue.get(fieldName));
}

// Example debug statement output:
// DEBUG|field name is Id, value is 001R0000003EPPuIAO
// DEBUG|field name is Name, value is TestMapAccount3
```

**Versioned Behavior Changes**

In API version 39.0 and later, `getPopulatedFieldsAsMap` returns all values set on the SObject, even if values were set after the record was queried. This behavior is dependent on the version of the apex class calling this method and not on the version of the class that generated the SObject. If you query an SObject at API version 20.0, and then call this method in a class with API version 40.0, you will get the full set of fields.

**getSObject** *(fieldName)*

Returns the value for the specified field. This method is primarily used with dynamic DML to access values for external IDs.

**Signature**

```java
public SObject getSObject(String fieldName)
```

**Parameters**

`fieldName`

Type: `String`

**Return Value**

Type: `SObject`
Example

Account acc = new account(Name = 'Acme', Description = 'Acme Account');
insert acc;
Contact con = new Contact(Lastname = 'AcmeCon', AccountId = acc.id);
insert con;

SObject contactDB =
[SELECT Id, AccountId, Account.Name FROM Contact WHERE id = :con.id LIMIT 1];
Account a = (Account)contactDB.getSObject('Account');
System.assertEquals('Acme', a.name);

**getSObject(field)**

Returns the value for the field specified by the field token `Schema.sObjectField` such as, `Schema.MyObj.MyExternalId`. This method is primarily used with dynamic DML to access values for external IDs.

**Signature**

```java
public SObject getSObject(Schema.SObjectField field)
```

**Parameters**

`field`

Type: `Schema.SObjectField`

**Return Value**

Type: `SObject`

**Usage**

If the method references polymorphic fields, a `Name object` is returned. Use the `TYPEOF` clause in the SOQL SELECT statement to directly get results that depend on the runtime object type referenced by the polymorphic field. See [Working with Polymorphic Relationships in SOQL Queries](#).

Example

Account acc = new account(name = 'Acme', description = 'Acme Account');
insert acc;
Contact con = new contact(lastname = 'AcmeCon', accountid = acc.id);
insert con;

Schema.DescribeFieldResult fieldResult = Contact.AccountId.getDescribe();
Schema.SObjectField field = fieldResult.getSObjectField();

SObject contactDB =
[SELECT Id, AccountId, Account.Name FROM Contact WHERE id = :con.id LIMIT 1];
Account a = (Account)contactDB.getSObject(field);
System.assertEquals('Acme', a.name);
**getSObjects(fieldName)**

Returns the values for the specified field. This method is primarily used with dynamic DML to access values for associated objects, such as child relationships.

**Signature**

```
public SObject[] getSObjects(String fieldName)
```

**Parameters**

- `fieldName`
  
  Type: String

**Return Value**

Type: SObject[]

**Usage**

For more information, see Dynamic DML.

**Example**

```
Account acc = new account(name = 'Acme', description = 'Acme Account');
insert acc;
Contact con = new contact(lastname = 'AcmeCon', accountid = acc.id);
insert con;
SObject[] a = [SELECT id, (SELECT Name FROM Contacts LIMIT 1) FROM Account WHERE id = :acc.id];
SObject[] contactsDB = a.get(0).getSObjects('Contacts');
String fieldValue = (String)contactsDB.get(0).get('Name');
System.assertEquals('AcmeCon', fieldValue);
```

---

**getSObjects(fieldToken)**

Returns the value for the field specified by the field token `Schema.fieldName`, such as, `Schema.Account.Contact`. This method is primarily used with dynamic DML to access values for associated objects, such as child relationships.

**Signature**

```
public SObject[] getSObjects(Schema.SObjectType fieldName)
```

**Parameters**

- `fieldName`
  
  Type: Schema.SObjectType

**Return Value**

Type: SObject[]
**getSObjectType()**

Returns the token for this SObject. This method is primarily used with describe information.

**Signature**

```java
public Schema.SObjectType getSObjectType()
```

**Return Value**

Type: `Schema.SObjectType`

**Usage**

For more information, see `apex_dynamic_describe_objects_understanding`.

**Example**

```java
Account acc = new Account(name = 'Acme', description = 'Acme Account');
Schema.SObjectType expected = Schema.Account.getSObjectType();
System.assertEquals(expected, acc.getSObjectType());
```

**getQuickActionName()**

Retrieves the name of a quick action associated with this SObject. Typically used in triggers.

**Signature**

```java
public String getQuickActionName()
```

**Return Value**

Type: `String`

**Example**

```java
trigger accTrig2 on Contact (before insert) {
    for (Contact c : Trigger.new) {
        if (c.getQuickActionName() == QuickAction.CreateContact) {
            c.WhereFrom__c = 'GlobaActionl';
        } else if (c.getQuickActionName() == Schema.Account.QuickAction.CreateContact) {
            c.WhereFrom__c = 'AccountAction';
        } else if (c.getQuickActionName() == null) {
            c.WhereFrom__c = 'NoAction';
        } else {
            System.assert(false);
        }
    }
}
```
hasErrors()

Returns true if an SObject instance has associated errors. The error message can be associated to the SObject instance by using SObject.addError(), validation rules, or by other means.

Signature

```
public Boolean hasErrors()
```

Return Value

Type: Boolean

isClone()

Returns true if an entity is cloned from something, even if the entity hasn’t been saved. The method can only be used within the transaction where the entity is cloned, as clone information doesn’t persist in subsequent transactions.

Signature

```
public Boolean isClone()
```

Return Value

Type: Boolean

Example

```
Account acc = new Account(Name = 'Acme');
insert acc;
Account acc2 = acc.clone();
// Test before saving
System.assertEquals(true, acc2.isClone());
insert acc2;
// Test after saving
System.assertEquals(true, acc2.isClone());
```

isSet(fieldName)

Returns information about the queried sObject field. Returns true if the sObject field is populated, either by direct assignment or by inclusion in a SOQL query. Returns false if the sObject field is not set. If an invalid field is specified, an SObjectException is thrown.

Signature

```
public Void isSet(String fieldName)
```

Parameters

```
fieldName
 Type: String
```
Return Value
Type: Boolean

Usage
The `isSet` method doesn't check if a field is accessible to a specific user via org permissions or other specialized access permissions.

Example
```
Contact c = new Contact(LastName = 'Joyce');
System.assertEquals(true, c.isSet('LastName'));
System.assertEquals(false, c.isSet('FirstName')); // FirstName field is not written to
System.assertEquals(true, c.isSet('FirstName')); // FirstName field is written to
```

`isSet(field)`
Returns information about the queried sObject field. Returns `true` if the sObject field is populated, either by direct assignment or by inclusion in a SOQL query. Returns `false` if the sObject field is not set. If an invalid field is specified, an SObjectException is thrown.

Signature
```
public Void isSet(Schema.SObjectField field)
```

Parameters
`field`
Type: `SObjectField Class`

Return Value
Type: Boolean

Usage
The `isSet` method doesn't check if a field is accessible to a specific user via org permissions or other specialized access permissions.

Example
```
Contact newContact = new Contact(LastName = 'Joyce');
insert(newContact); // Insert a new contact with last name Joyce
Contact c = [SELECT FirstName FROM Contact WHERE Id = :newContact.Id];
System.assertEquals(true, c.isSet(Contact.FirstName)); // FirstName field in query
System.assertEquals(false, c.isSet(Contact.LastName)); // LastName field not in query
```

`put(fieldName, value)`
Sets the value for the specified field and returns the previous value for the field.
**Signature**

```java
public Object put(String fieldName, Object value)
```

**Parameters**

- **fieldName**
  - Type: `String`
- **value**
  - Type: `Object`

**Return Value**

Type: `Object`

**Example**

```java
Account acc = new Account(name = 'test', description = 'old desc');
String oldDesc = (String)acc.put('description', 'new desc');
System.assertEquals('old desc', oldDesc);
System.assertEquals('new desc', acc.description);
```

**put(field, value)**

Sets the value for the field specified by the field token `Schema.sObjectField`, such as, `Schema.Account.AccountNumber` and returns the previous value for the field.

**Signature**

```java
public Object put(Schema.SObjectField field, Object value)
```

**Parameters**

- **field**
  - Type: `Schema.SObjectField`
- **value**
  - Type: `Object`

**Return Value**

Type: `Object`

**Example**

```java
Account acc = new Account(name = 'test', description = 'old desc');
String oldDesc = (String)acc.put(Schema.Account.Description, 'new desc');
System.assertEquals('old desc', oldDesc);
System.assertEquals('new desc', acc.description);
```
Note: Field tokens aren't available for person accounts. If you access Schema.Account.\textit{fieldName}, you get an exception error. Instead, specify the field name as a string.

\texttt{\textbf{putSObject(fieldName, value)}}

Sets the value for the specified field. This method is primarily used with dynamic DML for setting external IDs. The method returns the previous value of the field.

\textbf{Signature}

\begin{verbatim}
public SObject putSObject(String fieldName, SObject value)
\end{verbatim}

\textbf{Parameters}

\begin{itemize}
  \item \textit{fieldName}
    \begin{itemize}
      \item Type: String
    \end{itemize}
  \item \textit{value}
    \begin{itemize}
      \item Type: SObject
    \end{itemize}
\end{itemize}

\textbf{Return Value}

Type: SObject

\textbf{Example}

\begin{verbatim}
Account acc = new Account(name = 'Acme', description = 'Acme Account');
    insert acc;
Contact con = new contact(lastname = 'AcmeCon', accountid = acc.id);
    insert con;
Account acc2 = new account(name = 'Not Acme');

    Contact contactDB =
        (Contact)[SELECT Id, AccountId, Account.Name FROM Contact WHERE id = :con.id LIMIT 1];
    Account a = (Account)contactDB.putSObject('Account', acc2);
    System.assertEquals('Acme', a.name);
    System.assertEquals('Not Acme', contactDB.Account.name);
\end{verbatim}

\texttt{\textbf{putSObject(fieldName, value)}}

Sets the value for the field specified by the token Schema.SObjectType. This method is primarily used with dynamic DML for setting external IDs. The method returns the previous value of the field.

\textbf{Signature}

\begin{verbatim}
public SObject putSObject(Schema.SObjectType fieldName, SObject value)
\end{verbatim}

\textbf{Parameters}

\begin{itemize}
  \item \textit{fieldName}
    \begin{itemize}
      \item Type: Schema.SObjectType
    \end{itemize}
\end{itemize}
value
  Type: SObject

Return Value
Type: SObject

recalculateFormulas()
Deprecated as of API version 57.0. Use the recalculateFormulas() method in the System.Formula class instead.

Signature
public Void recalculateFormulas()

Return Value
Type: Void

Usage
This method doesn’t recalculate cross-object formulas. If you call this method on objects that have both cross-object and non-cross-object formula fields, only the non-cross-object formula fields are recalculated.

Each recalculateFormulas call counts against the SOQL query limits. See Execution Governors and Limits.

SEE ALSO:
  recalculateFormulas(sobjects)
  What Is a Cross-Object Formula?

setOptions(DMLOptions)
Sets the DMLOptions object for the SObject.

Signature
public Void setOptions(database.DMLOptions DMLOptions)

Parameters
  DMLOptions
    Type: Database.DMLOptions

Return Value
Type: Void
Example

```java
Database.DMLOptions dmo = new Database.dmlOptions();
dmo.assignmentRuleHeader.useDefaultRule = true;

Account acc = new Account(Name = 'Acme');
acc.setOptions(dmo);
```

**SObjectAccessDecision Class**

Contains the results of a call to the `Security.stripInaccessible` method and methods to retrieve those results.

**Namespace**

*System*

**IN THIS SECTION:**

SObjectAccessDecision Methods

**SObjectAccessDecision Methods**

The following are methods for `SObjectAccessDecision`.

**IN THIS SECTION:**

- `getModifiedIndexes()`
  - Returns the indexes of sObjects that are modified by the `stripInaccessible` method.
- `getRecords()`
  - Returns a list of new sObjects that are identical to the source records, except that they are stripped of fields that fail the field-level security check for the current user.
- `getRemovedFields()`
  - Returns a map of sObject types to their corresponding inaccessible fields. The map key is a string representation of the sObject type. The map value is a set of strings, which denote the fields names that are inaccessible.

**getModifiedIndexes()**

Returns the indexes of sObjects that are modified by the `stripInaccessible` method.

**Signature**

```java
public Set<Integer> getModifiedIndexes()
```

**Return Value**

Type: `Set<Integer>`

A set of unsigned integers that represent the row indexes of the modified sObjects.
Example

In this example, the user doesn’t have permission to update the `AnnualRevenue` field of an `Account`.

```java
List<Account> accounts = new List<Account>{{
    new Account(Name='Account1', AnnualRevenue=1000),
    new Account(Name='Account2')
}};

// Strip fields that are not updatable
SObjectAccessDecision decision = Security.stripInaccessible(
   AccessType.UPDATABLE,
    accounts);

// Print stripped records
for (SObject strippedAccount : decision.getRecords()) {
    System.debug(strippedAccount);
}

// Print modified indexes
System.debug(decision.getModifiedIndexes());
```

### getRecords()

Returns a list of new sObjects that are identical to the source records, except that they are stripped of fields that fail the field-level security check for the current user.

### Usage

The `stripInaccessible` method performs field-level access check for the source records in the context of the current user’s operation. The `getRecords()` method returns the new records that contain only the fields that the current user has access to.

### Signature

```java
public List<SObject> getRecords()
```

### Return Value

Type: `List<SObject>`

Even if the result list contains only one sObject, the return type is still a list (of size one).

### Example

In this example, the user doesn’t have permission to update the `AnnualRevenue` field of an `Account`.

```java
List<Account> accounts = new List<Account>{{
    new Account(Name='Account1', AnnualRevenue=1000),
    new Account(Name='Account2')
}};

// Strip fields that are not updatable
SObjectAccessDecision decision = Security.stripInaccessible(
   AccessType.UPDATABLE,
```
accounts;

// Print stripped records
for (SObject strippedAccount : decision.getRecords()) {
    System.debug(strippedAccount);
}

getRemovedFields()

Returns a map of sObject types to their corresponding inaccessible fields. The map key is a string representation of the sObject type. The map value is a set of strings, which denote the fields names that are inaccessible.

**Signature**

```
public Map<String,Set<String>> getRemovedFields()
```

**Return Value**

Type: `Map<String,Set<String>>`

**Example**

In this example, the user doesn’t have permission to update the `AnnualRevenue` field of an Account.

```
List<Account> accounts = new List<Account>{
    new Account(Name='Account1', AnnualRevenue=1000),
    new Account(Name='Account2')
};

// Strip fields that are not updatable
SObjectAccessDecision decision = Security.stripInaccessible(
    AccessType.UPDATABLE,
    accounts);

// Print stripped records
for (SObject strippedAccount : decision.getRecords()) {
    System.debug(strippedAccount);
}
// Print removed fields
System.debug(decision.getRemovedFields());
```

**StaticResourceCalloutMock Class**

Utility class used to specify a fake response for testing HTTP callouts.

**Namespace**

```
System
```
**Usage**

Use the methods in this class to set the response properties for testing HTTP callouts.

**StaticResourceCalloutMock Constructors**

The following are constructors for StaticResourceCalloutMock.

**StaticResourceCalloutMock()**

Creates a new instance of the StaticResourceCalloutMock class.

**Signature**

```
public StaticResourceCalloutMock()
```

**StaticResourceCalloutMock Methods**

The following are methods for StaticResourceCalloutMock. All are instance methods.

**setHeader(headerName, headerValue)**

Sets the specified header name and value for the fake response.

**setStaticResource(resourceName)**

Sets the specified static resource, which contains the response body.

**setStatus(httpStatus)**

Sets the specified HTTP status for the response.

**setStatusCode(httpStatusCode)**

Sets the specified HTTP status for the response.

**setHeader(headerName, headerValue)**

Sets the specified header name and value for the fake response.

**Signature**

```
public Void setHeader(String headerName, String headerValue)
```
Parameters

headerName
  Type: String

headerValue
  Type: String

Return Value
  Type: Void

**setStaticResource(resourceName)**
Sets the specified static resource, which contains the response body.

Signature

```java
public Void setStaticResource(String resourceName)
```

Parameters

resourceName
  Type: String

Return Value
  Type: Void

**setStatus(httpStatus)**
Sets the specified HTTP status for the response.

Signature

```java
public Void setStatus(String httpStatus)
```

Parameters

httpStatus
  Type: String

Return Value
  Type: Void

**getStatusCode(httpStatusCode)**
Sets the specified HTTP status for the response.
Signature

```java
public Void setStatusCode(Integer httpStatusCode)
```

Parameters

`httpStatusCode`
Type: Integer

Return Value

Type: Void

String Class
Contains methods for the String primitive data type.

Namespace

`System`

Usage

For more information on Strings, see String Data Type.

String Methods

The following are methods for String.

IN THIS SECTION:

- `abbreviate(maxWidth)`
  Returns an abbreviated version of the String, of the specified length and with ellipses appended if the current String is longer than the specified length; otherwise, returns the original String without ellipses.

- `abbreviate(maxWidth, offset)`
  Returns an abbreviated version of the String, starting at the specified character offset and of the specified length. The returned String has ellipses appended at the start and the end if characters have been removed at these locations.

- `capitalize()`
  Returns the current String with the first letter changed to title case.

- `center(size)`
  Returns a version of the current String of the specified size padded with spaces on the left and right, so that it appears in the center. If the specified size is smaller than the current String size, the entire String is returned without added spaces.

- `center(size, paddingString)`
  Returns a version of the current String of the specified size padded with the specified String on the left and right, so that it appears in the center. If the specified size is smaller than the current String size, the entire String is returned without padding.

- `charAt(index)`
  Returns the value of the character at the specified index.
codePointAt(index)
Returns the Unicode code point value at the specified index.

codePointBefore(index)
Returns the Unicode code point value that occurs before the specified index.

codePointCount(beginIndex, endIndex)
Returns the number of Unicode code points within the specified text range.

compareTo(secondString)
Compares two strings lexicographically, based on the Unicode value of each character in the Strings.

contains(substring)
Returns true if and only if the String that called the method contains the specified sequence of characters in substring.

containsAny(inputString)
Returns true if the current String contains any of the characters in the specified String; otherwise, returns false.

containsIgnoreCase(substring)
Returns true if the current String contains the specified sequence of characters without regard to case; otherwise, returns false.

containsNone(inputString)
Returns true if the current String doesn’t contain any of the characters in the specified String; otherwise, returns false.

containsOnly(inputString)
Returns true if the current String contains characters only from the specified sequence of characters and not any other characters; otherwise, returns false.

containsWhitespace()
Returns true if the current String contains any white space characters; otherwise, returns false.

countMatches(substring)
Returns the number of times the specified substring occurs in the current String.

deleteWhitespace()
Returns a version of the current String with all white space characters removed.

difference(secondString)
Returns the difference between the current String and the specified String.

endsWith(suffix)
Returns true if the String that called the method ends with the specified suffix.

endsWithIgnoreCase(suffix)
Returns true if the current String ends with the specified suffix; otherwise, returns false.

equals(secondString)
Deprecated. This method is replaced by equals(stringOrId). Returns true if the passed-in string is not null and represents the same binary sequence of characters as the current string. Use this method to perform case-sensitive comparisons.

equals(stringOrId)
Returns true if the passed-in object is not null and represents the same binary sequence of characters as the current string. Use this method to compare a string to an object that represents a string or an ID.

equalsIgnoreCase(secondString)
Returns true if the secondString isn’t null and represents the same sequence of characters as the String that called the method, ignoring case.
escapeCsv()
Returns a String for a CSV column enclosed in double quotes, if required.

escapeEcmaScript()
Escapes the characters in the String using EcmaScript String rules.

escapeHtml3()
Escapes the characters in a String using HTML 3.0 entities.

escapeHtml4()
Escapes the characters in a String using HTML 4.0 entities.

escapeJava()
Returns a String whose characters are escaped using Java String rules. Characters escaped include quotes and control characters, such as tab, backslash, and carriage return characters.

escapeSingleQuotes(stringToEscape)
Returns a String with the escape character (\) added before any single quotation marks in the String s.

escapeUnicode()
Returns a String whose Unicode characters are escaped to a Unicode escape sequence.

escapeXml()
Escapes the characters in a String using XML entities.

format(stringToFormat, formattingArguments)
Treat the first argument as a pattern and return a string using the second argument for substitution and formatting. The substitution and formatting are the same as apex:outputText and the Java MessageFormat class. Non-string types in the second argument’s List are implicitly converted to strings, respecting the toString() method overrides that exist on the type.

fromCharArray(charArray)
Returns a String from the values of the list of integers.

getChars()
Returns an array of character values that represent the characters in this string.

getCommonPrefix(strings)
Returns the initial sequence of characters as a String that is common to all the specified Strings.

getLevenshteinDistance(stringToCompare)
Returns the Levenshtein distance between the current String and the specified String.

getLevenshteinDistance(stringToCompare, threshold)
Returns the Levenshtein distance between the current String and the specified String if it is less than or equal than the given threshold; otherwise, returns -1.

hashCode()
Returns a hash code value for this string.

indexOf(substring)
Returns the index of the first occurrence of the specified substring. If the substring does not occur, this method returns -1.

indexOf(substring, index)
Returns the zero-based index of the first occurrence of the specified substring from the point of the given index. If the substring does not occur, this method returns -1.
indexOfAny(substring)
Returns the zero-based index of the first occurrence of any character specified in the substring. If none of the characters occur, returns -1.

indexOfAnyBut(substring)
Returns the zero-based index of the first occurrence of a character that is not in the specified substring. Otherwise, returns -1.

indexOfChar(character)
Returns the index of the first occurrence of the character that corresponds to the specified character value.

indexOfChar(character, startIndex)
Returns the index of the first occurrence of the character that corresponds to the specified character value, starting from the specified index.

indexOfDifference(stringToCompare)
Returns the zero-based index of the character where the current String begins to differ from the specified String.

indexOfIgnoreCase(substring)
Returns the zero-based index of the first occurrence of the specified substring without regard to case. If the substring does not occur, this method returns -1.

indexOfIgnoreCase(substring, startPosition)
Returns the zero-based index of the first occurrence of the specified substring from the point of index startPosition, without regard to case. If the substring does not occur, this method returns -1.

isAllLowerCase()
Returns true if all characters in the current String are lowercase; otherwise, returns false.

isAllUpperCase()
Returns true if all characters in the current String are uppercase; otherwise, returns false.

isAlpha()
Returns true if all characters in the current String are Unicode letters only; otherwise, returns false.

isAlphaSpace()
Returns true if all characters in the current String are Unicode letters or spaces only; otherwise, returns false.

isAlphanumeric()
Returns true if all characters in the current String are Unicode letters or numbers only; otherwise, returns false.

isAlphanumericSpace()
Returns true if all characters in the current String are Unicode letters, numbers, or spaces only; otherwise, returns false.

isAsciiPrintable()
Returns true if the current String contains only ASCII printable characters; otherwise, returns false.

isBlank(inputString)
Returns true if the specified String is white space, empty (""), or null; otherwise, returns false.

isEmpty(inputString)
Returns true if the specified String is empty (") or null; otherwise, returns false.

isNotBlank(inputString)
Returns true if the specified String is not whitespace, not empty (""), and not null; otherwise, returns false.

isNotEmpty(inputString)
Returns true if the specified String is not empty (") and not null; otherwise, returns false.
isNumeric() Returns true if the current String contains only Unicode digits; otherwise, returns false.

isNumericSpace() Returns true if the current String contains only Unicode digits or spaces; otherwise, returns false.

isWhitespace() Returns true if the current String contains only white space characters or is empty; otherwise, returns false.

join(iterableObj, separator) Joins the elements of the specified iterable object, such as a List, into a single String separated by the specified separator.

lastIndexOf(substring) Returns the index of the last occurrence of the specified substring. If the substring does not occur, this method returns -1.

lastIndexOf(substring, endPosition) Returns the index of the last occurrence of the specified substring, starting from the character at index 0 and ending at the specified index.

lastIndexOfChar(character) Returns the index of the last occurrence of the character that corresponds to the specified character value.

lastIndexOfChar(character, endIndex) Returns the index of the last occurrence of the character that corresponds to the specified character value, starting from the specified index.

lastIndexOfIgnoreCase(substring) Returns the index of the last occurrence of the specified substring regardless of case.

lastIndexOfIgnoreCase(substring, endPosition) Returns the index of the last occurrence of the specified substring regardless of case, starting from the character at index 0 and ending at the specified index.

left(length) Returns the leftmost characters of the current String of the specified length.

leftPad(length) Returns the current String padded with spaces on the left and of the specified length.

leftPad(length, padStr) Returns the current String padded with String padStr on the left and of the specified length.

length() Returns the number of 16-bit Unicode characters contained in the String.

mid(startIndex, length) Returns a new String that begins with the character at the specified zero-based startIndex with the number of characters specified by length.

normalizeSpace() Returns the current String with leading, trailing, and repeating white space characters removed.

offsetByCodePoints(index, codePointOffset) Returns the index of the Unicode code point that is offset by the specified number of code points, starting from the given index.

remove(substring) Removes all occurrences of the specified substring and returns the String result.
removeEnd(substring)
Removes the specified substring only if it occurs at the end of the String.

removeEndIgnoreCase(substring)
Removes the specified substring only if it occurs at the end of the String using a case-insensitive match.

removeStart(substring)
Removes the specified substring only if it occurs at the beginning of the String.

removeStartIgnoreCase(substring)
Removes the specified substring only if it occurs at the beginning of the String using a case-insensitive match.

repeat(numberOfTimes)
Returns the current String repeated the specified number of times.

repeat(separator, numberOfTimes)
Returns the current String repeated the specified number of times using the specified separator to separate the repeated Strings.

replace(target, replacement)
Replaces each substring of a string that matches the literal target sequence target with the specified literal replacement sequence replacement.

replaceAll(regExp, replacement)
Replaces each substring of a string that matches the regular expression regExp with the replacement sequence replacement.

replaceFirst(regExp, replacement)
Replaces the first substring of a string that matches the regular expression regExp with the replacement sequence replacement.

reverse()
Returns a String with all the characters reversed.

right(length)
Returns the rightmost characters of the current String of the specified length.

rightPad(length)
Returns the current String padded with spaces on the right and of the specified length.

rightPad(length, padStr)
Returns the current String padded with String padStr on the right and of the specified length.

split(regExp)
Returns a list that contains each substring of the String that is terminated by either the regular expression regExp or the end of the String.

split(regExp, limit)
Returns a list that contains each substring of the String that is terminated by either the regular expression regExp or the end of the String.

splitByCharacterType()
Splits the current String by character type and returns a list of contiguous character groups of the same type as complete tokens.

splitByCharacterTypeCamelCase()
Splits the current String by character type and returns a list of contiguous character groups of the same type as complete tokens, with the following exception: the uppercase character, if any, immediately preceding a lowercase character token belongs to the following character token rather than to the preceding.

startsWith(prefix)
Returns true if the String that called the method begins with the specified prefix.
startsWithIgnoreCase(prefix)
Returns true if the current String begins with the specified prefix regardless of the prefix case.

stripHtmlTags()
Removes HTML markup and returns plain text.

substring(startIndex)
Returns a new String that begins with the character at the specified zero-based startIndex and extends to the end of the String.

substring(startIndex, endIndex)
Returns a new String that begins with the character at the specified zero-based startIndex and extends to the character at endIndex - 1.

substringAfter(separator)
Returns the substring that occurs after the first occurrence of the specified separator.

substringAfterLast(separator)
Returns the substring that occurs after the last occurrence of the specified separator.

substringBefore(separator)
Returns the substring that occurs before the first occurrence of the specified separator.

substringBeforeLast(separator)
Returns the substring that occurs before the last occurrence of the specified separator.

substringBetween(tag)
Returns the substring that occurs between two instances of the specified tag String.

substringBetween(open, close)
Returns the substring that occurs between the two specified Strings.

swapCase()
Swaps the case of all characters and returns the resulting String by using the default (English US) locale.

toLowerCase()
Converts all of the characters in the String to lowercase using the rules of the default (English US) locale.

toLowerCase(locale)
Converts all of the characters in the String to lowercase using the rules of the specified locale.

toUpperCase()
Converts all of the characters in the String to uppercase using the rules of the default (English US) locale.

toUpperCase(locale)
Converts all of the characters in the String to uppercase using the rules of the specified locale.

trim()
Returns a copy of the string that no longer contains any leading or trailing white space characters.

uncapitalize()
Returns the current String with the first letter in lowercase.

unescapeCsv()
Returns a String representing an unescaped CSV column.

unescapeEcmaScript()
Unescapes any EcmaScript literals found in the String.
unescapeHtml3()  
Unescapes the characters in a String using HTML 3.0 entities.

unescapeHtml4()  
Unescapes the characters in a String using HTML 4.0 entities.

unescapeJava()  
Returns a String whose Java literals are unescaped. Literals unescaped include escape sequences for quotes (""), and control characters, such as tab (\t), and carriage return (\n).

unescapeUnicode()  
Returns a String whose escaped Unicode characters are unescaped.

unescapeXml()  
Unescapes the characters in a String using XML entities.

valueOf(dateToConvert)  
Returns a String that represents the specified Date in the standard “yyyy-MM-dd” format.

valueOf(datetimeToConvert)  
Returns a String that represents the specified Datetime in the standard “yyyyy-MM-dd HH:mm:ss” format for the local time zone.

valueOf(decimalToConvert)  
Returns a String that represents the specified Decimal.

valueOf(doubleToConvert)  
Returns a String that represents the specified Double.

valueOf(integerToConvert)  
Returns a String that represents the specified Integer.

valueOf(longToConvert)  
Returns a String that represents the specified Long.

valueOf(toConvert)  
Returns a string representation of the specified object argument.

valueOfGmt(datetimeToConvert)  
Returns a String that represents the specified Datetime in the standard “yyyy-MM-dd HH:mm:ss” format for the GMT time zone.

---

**abbreviate(maxWidth)**

Returns an abbreviated version of the String, of the specified length and with ellipses appended if the current String is longer than the specified length; otherwise, returns the original String without ellipses.

**Signature**

public String abbreviate(Integer maxWidth)

**Parameters**

*maxWidth*  
Type: ```Integer```  

If *maxWidth* is less than four, this method throws a run-time exception.
Return Value
Type: String

Example
```java
String s = 'Hello Maximillian';
String s2 = s.abbreviate(8);
System.assertEquals('Hello...', s2);
System.assertEquals(8, s2.length());
```

**abbreviate (maxWidth, offset)**
Returns an abbreviated version of the String, starting at the specified character offset and of the specified length. The returned String has ellipses appended at the start and the end if characters have been removed at these locations.

**Signature**
```java
public String abbreviate(Integer maxWidth, Integer offset)
```

**Parameters**
- **maxWidth**
  Type: Integer
  Note that the offset is not necessarily the leftmost character in the returned String or the first character following the ellipses, but it appears somewhere in the result. Regardless, abbreviate won't return a String of length greater than maxWidth. If maxWidth is too small, this method throws a run-time exception.

- **offset**
  Type: Integer

**Return Value**
Type: String

Example
```java
String s = 'Hello Maximillian';
// Start at M
String s2 = s.abbreviate(9,6);
System.assertEquals('...Max...', s2);
System.assertEquals(9, s2.length());
```

**capitalize()**
Returns the current String with the first letter changed to title case.

**Signature**
```java
public String capitalize()
```
Return Value
Type: String

Usage
This method is based on the Character.toTitleCase(char) Java method.

Example
```java
String s = 'hello maximillian';
String s2 = s.capitalize();
System.assertEquals('Hello maximillian', s2);
```

center(size)
Returns a version of the current String of the specified size padded with spaces on the left and right, so that it appears in the center. If the specified size is smaller than the current String size, the entire String is returned without added spaces.

Signature
public String center(Integer size)

Parameters
size
Type: Integer

Return Value
Type: String

Example
```java
String s = 'hello';
String s2 = s.center(9);
System.assertEquals(
   ' hello ',
   s2);
```

center(size, paddingString)
Returns a version of the current String of the specified size padded with the specified String on the left and right, so that it appears in the center. If the specified size is smaller than the current String size, the entire String is returned without padding.

Signature
public String center(Integer size, String paddingString)
Parameters

size
  Type: Integer

paddingString
  Type: String

Return Value
Type: String

Example

```java
String s = 'hello';
String s2 = s.center(9, '-');
System.assertEquals('--hello--', s2);
```

**charAt(index)**

Returns the value of the character at the specified index.

**Signature**

```java
public Integer charAt(Integer index)
```

**Parameters**

index
  Type: Integer
  The index of the character to get the value of.

**Return Value**

Type: Integer
  The integer value of the character.

**Usage**

The `charAt` method returns the value of the character pointed to by the specified index. If the index points to the beginning of a surrogate pair (the high-surrogate code point), this method returns only the high-surrogate code point. To return the supplementary code point corresponding to a surrogate pair, call `codePointAt` instead.

**Example**

This example gets the value of the first character at index 0.

```java
String str = 'Ω is Omega.';
System.assertEquals(937, str.charAt(0));
```
This example shows the difference between `charAt` and `codePointAt`. The example calls these methods on escaped supplementary Unicode characters. `charAt(0)` returns the high surrogate value, which corresponds to \uD835. `codePointAt(0)` returns the value for the entire surrogate pair.

```java
String str = '\uD835\uDD0A';
System.assertEquals(55349, str.charAt(0),
  'charAt(0) didn\'t return the high surrogate.');
System.assertEquals(120074, str.codePointAt(0),
  'codePointAt(0) didn\'t return the entire two-character supplementary value.');
```

codePointAt(index)

Returns the Unicode code point value at the specified index.

**Signature**

```java
public Integer codePointAt(Integer index)
```

**Parameters**

- `index`  
  Type: Integer  
  The index of the characters (Unicode code units) in the string. The index range is from zero to the string length minus one.

**Return Value**

- Type: Integer  
  The Unicode code point value at the specified index.

**Usage**

If the `index` points to the beginning of a surrogate pair (the high-surrogate code point), and the character value at the following index points to the low-surrogate code point, this method returns the supplementary code point corresponding to this surrogate pair. Otherwise, this method returns the character value at the given index.

For more information on Unicode and surrogate pairs, see [The Unicode Consortium](https://www.unicode.org/).

**Example**

This example gets the code point value of the first character at index 0, which is the escaped Omega character. Also, the example gets the code point at index 20, which corresponds to the escaped supplementary Unicode characters (a pair of characters). Finally, it verifies that the escaped and unescaped forms of Omega have the same code point values.

The supplementary characters in this example (\uD835\uDD0A) correspond to mathematical fraktur capital G: \u03A9

```java
String str = '\u03A9 is Ω (Omega), and \uD835\uDD0A \n  is Fraktur Capital G..GeneratedValue=';
System.assertEquals(937, str.codePointAt(0));
System.assertEquals(120074, str.codePointAt(20));
// Escaped or unescaped forms of the same character have the same code point
System.assertEquals(str.codePointAt(0), str.codePointAt(5));
```
**codePointBefore(index)**

Returns the Unicode code point value that occurs before the specified index.

**Signature**

```java
public Integer codePointBefore(Integer index)
```

**Parameters**

*index*

Type: `Integer`

The index before the Unicode code point that is to be returned. The index range is from one to the string length.

**Return Value**

Type: `Integer`

The character or Unicode code point value that occurs before the specified index.

**Usage**

If the character value at index -1 is the low-surrogate code point, and index -2 is not negative and the character at this index location is the high-surrogate code point, this method returns the supplementary code point corresponding to this surrogate pair. If the character value at index -1 is an unpaired low-surrogate or high-surrogate code point, the surrogate value is returned.

For more information on Unicode and surrogate pairs, see The Unicode Consortium.

**Example**

This example gets the code point value of the first character (before index 1), which is the escaped Omega character. Also, the example gets the code point at index 20, which corresponds to the escaped supplementary characters (the two characters before index 22).

```java
String str = '\u03A9 is Ω (Omega), and \uD835\uDD0A ' +
            ' is Fraktur Capital G.';
System.assertEquals(937, str.codePointBefore(1));
System.assertEquals(120074, str.codePointBefore(22));
```

**codePointCount(beginIndex, endIndex)**

Returns the number of Unicode code points within the specified text range.

**Signature**

```java
public Integer codePointCount(Integer beginIndex, Integer endIndex)
```

**Parameters**

*beginIndex*

Type: `Integer`

The index of the first character in the range.
endIndex
Type: Integer
The index after the last character in the range.

Return Value
Type: Integer
The number of Unicode code points within the specified range.

Usage
The specified range begins at beginIndex and ends at endIndex–1. Unpaired surrogates within the text range count as one code point each.

Example
This example writes the count of code points in a substring that contains an escaped Unicode character and another substring that contains Unicode supplementary characters, which count as one code point.

```java
String str = '\u03A9 and \uD835\uDD0A characters.\';
System.debug('Count of code points for ' + str.substring(0,1) + ': ' + str.codePointCount(0,1));
System.debug('Count of code points for ' + str.substring(6,8) + ': ' + str.codePointCount(6,8));
```

// Output:
// Count of code points for Ω: 1
// Count of code points for ☯: 1

compareTo(secondString)
Compares two strings lexicographically, based on the Unicode value of each character in the Strings.

Signature
public Integer compareTo(String secondString)

Parameters
secondString
Type: String

Return Value
Type: Integer

Usage
The result is:
- A negative Integer if the String that called the method lexicographically precedes secondString
- A positive Integer if the String that called the method lexicographically follows `compsecondStringString`
- Zero if the Strings are equal

If there is no index position at which the Strings differ, then the shorter String lexicographically precedes the longer String.

Note that this method returns 0 whenever the `equals` method returns true.

**Example**

```java
String myString1 = 'abcde';
String myString2 = 'abcd';
Integer result =
    myString1.compareTo(myString2);
System.assertEquals(result, 1);
```

### contains(substring)

Returns `true` if and only if the String that called the method contains the specified sequence of characters in `substring`.

**Signature**

```java
public Boolean contains(String substring)
```

**Parameters**

- **substring**
  - Type: `String`

**Return Value**

- Type: `Boolean`

**Example**

```java
String myString1 = 'abcde';
String myString2 = 'abcd';
Boolean result =
    myString1.contains(myString2);
System.assertEquals(result, true);
```

### containsAny(inputString)

Returns `true` if the current String contains any of the characters in the specified String; otherwise, returns `false`.

**Signature**

```java
public Boolean containsAny(String inputString)
```

**Parameters**

- **inputString**
  - Type: `String`
Return Value
Type: Boolean

Example

```java
String s = 'hello';
Boolean b1 = s.containsAny('hx');
Boolean b2 = s.containsAny('x');
System.assertEquals(true, b1);
System.assertEquals(false, b2);
```

containsIgnoreCase(substring)
Returns true if the current String contains the specified sequence of characters without regard to case; otherwise, returns false.

Signature

```java
public Boolean containsIgnoreCase(String substring)
```

Parameters

substring
Type: String

Return Value
Type: Boolean

Example

```java
String s = 'hello';
Boolean b = s.containsIgnoreCase('HE');
System.assertEquals(true, b);
```

containsNone(inputString)
Returns true if the current String doesn’t contain any of the characters in the specified String; otherwise, returns false.

Signature

```java
public Boolean containsNone(String inputString)
```

Parameters

inputString
Type: String

If inputString is an empty string or the current String is empty, this method returns true. If inputString is null, this method returns a run-time exception.
Return Value
Type: Boolean

Example
```java
String s1 = 'abcde';
System.assert(s1.containsNone('fg'));
```

**containsOnly**(inputString)
Returns `true` if the current String contains characters only from the specified sequence of characters and not any other characters; otherwise, returns `false`.

Signature
```java
public Boolean containsOnly(String inputString)
```

Parameters
- `inputString`
  Type: String

Return Value
Type: Boolean

Example
```java
String s1 = 'abba';
String s2 = 'abba xyz';
Boolean b1 =
    s1.containsOnly('abcd');
System.assertEquals(
    true,
    b1);
Boolean b2 =
    s2.containsOnly('abcd');
System.assertEquals(
    false,
    b2);
```

**containsWhitespace()**
Returns `true` if the current String contains any white space characters; otherwise, returns `false`.

Signature
```java
public Boolean containsWhitespace()
```
Return Value
Type: Boolean

Example
```java
String s = 'Hello Jane';
System.assert(s.containsWhitespace()); //true
s = 'HelloJane ';
System.assert(s.containsWhitespace()); //true
s = 'HelloJane';
System.assert(s.containsWhitespace()); //true
s = 'HelloJane';
System.assert(!s.containsWhitespace()); //false
```

countMatches(substring)
Returns the number of times the specified substring occurs in the current String.

Signature
```java
public Integer countMatches(String substring)
```

Parameters
substring
Type: String

Return Value
Type: Integer

Example
```java
String s = 'Hello Jane';
System.assertEquals(1, s.countMatches('Hello'));
s = 'Hello Hello';
System.assertEquals(2, s.countMatches('Hello'));
s = 'Hello hello';
System.assertEquals(1, s.countMatches('Hello'));
```

deleteWhitespace()
Returns a version of the current String with all white space characters removed.

Signature
```java
public String deleteWhitespace()
```

Return Value
Type: String
Example

```java
String s1 = 'Hello Jane';
String s2 = 'HelloJane';
System.assertEquals(s2, s1.deleteWhitespace());
```

difference(secondString)

Returns the difference between the current String and the specified String.

Signature

```java
public String difference(String secondString)
```

Parameters

secondString

Type: String

- If `secondString` is an empty string, this method returns an empty string.
- If `secondString` is null, this method throws a run-time exception.

Return Value

Type: String

Example

```java
String s = 'Hello Jane';
String d1 = s.difference('Hello Max');
System.assertEquals('Max', d1);
String d2 = s.difference('Goodbye');
System.assertEquals('Goodbye', d2);
```

endsWith(suffix)

Returns `true` if the String that called the method ends with the specified `suffix`.

Signature

```java
public Boolean endsWith(String suffix)
```

Parameters

suffix

Type: String
Return Value
Type: Boolean

Example
```java
String s = 'Hello Jason';
System.assert(s.endsWith('Jason'));
```

`endsWithIgnoreCase(suffix)`

Returns `true` if the current String ends with the specified suffix; otherwise, returns `false`.

Signature
```java
public Boolean endsWithIgnoreCase(String suffix)
```

Parameters
- `suffix`
  Type: String

Return Value
Type: Boolean

Example
```java
String s = 'Hello Jason';
System.assert(s.endsWithIgnoreCase('jason'));
```

`equals(secondString)`

Deprecated. This method is replaced by `equals(stringOrId)`. Returns `true` if the passed-in string is not null and represents the same binary sequence of characters as the current string. Use this method to perform case-sensitive comparisons.

Signature
```java
public Boolean equals(String secondString)
```

Parameters
- `secondString`
  Type: String

Return Value
Type: Boolean
Usage

This method returns `true` when the `compareTo` method returns 0.

Use this method to perform case-sensitive comparisons. In contrast, the `==` operator performs case-insensitive string comparisons to match Apex semantics.

Example

```java
String myString1 = 'abcde';
String myString2 = 'abcd';
Boolean result = myString1.equals(myString2);
System.assertEquals(result, false);
```

`equals(stringOrId)`

Returns `true` if the passed-in object is not null and represents the same binary sequence of characters as the current string. Use this method to compare a string to an object that represents a string or an ID.

Signature

```java
public Boolean equals(Object stringOrId)
```

Parameters

`stringOrId`

Type: Object

Return Value

Type: Boolean

Usage

If you compare ID values, the lengths of IDs don't need to be equal. For example, if you compare a 15-character ID string to an object that represents the equivalent 18-character ID value, this method returns `true`. For more information about 15-character and 18-character IDs, see the ID Data Type.

Use this method to perform case-sensitive comparisons. In contrast, the `==` operator performs case-insensitive string comparisons to match Apex semantics.

Example

These examples show comparisons between different types of variables with both equal and unequal values. The examples also show how Apex automatically converts certain values before comparing them.

```java
// Compare a string to an object containing a string
Object obj1 = 'abc';
String str = 'abc';
Boolean result1 = str.equals(obj1);
System.assertEquals(result1, true);

// Compare a string to an object containing a number
```
```java
Integer obj2 = 100;
Boolean result2 = str.equals(obj2);
System.assertEquals(false, result2);

// Compare a string to an ID of the same length.
// 15-character ID
Id idValue15 = '001D000000Ju1zH';
// 15-character ID string value
String stringValue15 = '001D000000Ju1zH';
Boolean result3 = stringValue15.equals(IdValue15);
System.assertEquals(true, result3);

// Compare two equal ID values of different lengths:
// 15-character ID and 18-character ID
Id idValue18 = '001D000000Ju1zHIAR';
Boolean result4 = stringValue15.equals(IdValue18);
System.assertEquals(true, result4);
```

**equalsIgnoreCase(secondString)**

Returns `true` if the `secondString` isn’t null and represents the same sequence of characters as the String that called the method, ignoring case.

**Signature**

```java
public Boolean equalsIgnoreCase(String secondString)
```

**Parameters**

- **secondString**
  
  Type: **String**

**Return Value**

Type: **Boolean**

**Usage**

The `String.equalsIgnoreCase()` method ignores the locale of the context user. If you want the string comparison to be performed according to the locale, use the `==` operator instead. The `String.equalsIgnoreCase()` method typically executes faster than the operator because the method ignores the locale.

**Example**

```java
String myString1 = 'abcd';
String myString2 = 'ABCD';
Boolean result =
myString1.equalsIgnoreCase(myString2);
System.assertEquals(result, true);
```
**escapeCsv()**

Returns a String for a CSV column enclosed in double quotes, if required.

**Signature**

```java
public String escapeCsv()
```

**Return Value**

Type: String

**Usage**

If the String contains a comma, newline or double quote, the returned String is enclosed in double quotes. Also, any double quote characters in the String are escaped with another double quote.

If the String doesn’t contain a comma, newline or double quote, it is returned unchanged.

**Example**

```java
String s1 = 'Max1, "Max2"';
String s2 = s1.escapeCsv();
System.assertEquals("Max1, "Max2"", s2);
```

**escapeEcmaScript()**

Escapes the characters in the String using EcmaScript String rules.

**Signature**

```java
public String escapeEcmaScript()
```

**Return Value**

Type: String

**Usage**

The only difference between Apex strings and EcmaScript strings is that in EcmaScript, a single quote and forward-slash (/) are escaped.

**Example**

```java
String s1 = "grade": 3.9/4.0;
String s2 = s1.escapeEcmaScript();
System.debug(s2);
// Output is:
// "grade": 3.9\/4.0
System.assertEquals("'\"grade\": 3.9\/4.0', s2);
```
**escapeHtml3()**
Escapes the characters in a String using HTML 3.0 entities.

**Signature**

```java
public String escapeHtml3()
```

**Return Value**
Type: **String**

**Example**

```java
String s1 = '
"<Black&White>"';
String s2 = s1.escapeHtml3();
System.debug(s2);
// Output:
// &quot;&lt;Black&amp;
// White&gt;&quot;
```

**escapeHtml4()**
Escapes the characters in a String using HTML 4.0 entities.

**Signature**

```java
public String escapeHtml4()
```

**Return Value**
Type: **String**

**Example**

```java
String s1 = '
"<Black&White>"';
String s2 = s1.escapeHtml4();
System.debug(s2);
// Output:
// &quot;&lt;Black&amp;
// White&gt;&quot;
```

**escapeJava()**
Returns a String whose characters are escaped using Java String rules. Characters escaped include quotes and control characters, such as tab, backslash, and carriage return characters.
Apex Reference Guide

String Class

Signature
public String escapeJava()

Return Value
Type: String
The escaped string.

Example
// Input string contains quotation marks
String s = 'Company: "Salesforce.com"';
String escapedStr = s.escapeJava();
// Output string has the quotes escpaded
System.assertEquals('Company: \\"Salesforce.com\\"', escapedStr);

escapeSingleQuotes(stringToEscape)
Returns a String with the escape character (\) added before any single quotation marks in the String s.

Signature
public static String escapeSingleQuotes(String stringToEscape)

Parameters
stringToEscape

Type: String

Return Value
Type: String

Usage
This method is useful when creating a dynamic SOQL statement, to help prevent SOQL injection. For more information on dynamic
SOQL, see Dynamic SOQL.

Example
String s = '\'Hello Jason\'';
system.debug(s); // Outputs 'Hello Jason'
String escapedStr = String.escapeSingleQuotes(s);
// Outputs \'Hello Jason\'
system.debug(escapedStr);
// Escapes the string \\\' to string \'
system.assertEquals('\\\'Hello Jason\\\'', escapedStr);

3426


**escapeUnicode()**
Returns a String whose Unicode characters are escaped to a Unicode escape sequence.

**Signature**
```
public String escapeUnicode()
```

**Return Value**
Type: `String`
The escaped string.

**Example**
```
String s = 'De onde você é?';
String escapedStr = s.escapeUnicode();
System.assertEquals('De onde voc\u00E9?', escapedStr);
```

**escapeXml()**
Escapes the characters in a String using XML entities.

**Signature**
```
public String escapeXml()
```

**Return Value**
Type: `String`

**Usage**
Supports only the five basic XML entities (gt, lt, quot, amp, apos). Does not support DTDs or external entities. Unicode characters greater than 0x7f are not escaped.

**Example**
```
String s1 = "<Black&White>";
String s2 = s1.escapeXml();
System.debug(s2);
// Output:
// &quot;&lt;Black&amp;White&gt;";
```
format(stringToFormat, formattingArguments)

Treat the first argument as a pattern and return a string using the second argument for substitution and formatting. The substitution and formatting are the same as apex:outputText and the Java MessageFormat class. Non-string types in the second argument’s List are implicitly converted to strings, respecting the toString() method overrides that exist on the type.

Signature

public static String format(String stringToFormat, List<Object> formattingArguments)

Parameters

stringToFormat
Type: String

formattingArguments
Type: List<Object>

Return Value
Type: String

Versioned Behavior Changes

From version 51.0 and later, the format() method supports single quotes in the stringToFormat parameter and returns a formatted string using the formattingArguments parameter. In version 50.0 and earlier, single quotes weren’t supported.

Example

String template = '{0} was last updated {1}';
List<Object> parameters = new List<Object> {'Universal Containers',
DateTime.newInstance(2018, 11, 15) };
String formatted = String.format(template, parameters);
System.debug ('Newly formatted string is:' + formatted);

fromCharArray(charArray)

Returns a String from the values of the list of integers.

Signature

public static String fromCharArray(List<Integer> charArray)

Parameters

charArray
Type: List<Integer>

Return Value
Type: String
Example

```java
List<Integer> charArr = new Integer[]{74};
String convertedChar = String.fromCharCode(charArr);
System.assertEquals('J', convertedChar);
```

getChars()

Returns an array of character values that represent the characters in this string.

Signature

```java
public List<Integer> getChars()
```

Return Value

Type: `List<Integer>`

A list of integers, each corresponding to a character value in the string.

Example

This sample converts a string to a character array and then gets the first array element, which corresponds to the value of 'J'.

```java
String str = 'Jane goes fishing.';
Integer[] chars = str.getChars();
// Get the value of 'J'
System.assertEquals(74, chars[0]);
```

getCommonPrefix(strings)

Returns the initial sequence of characters as a String that is common to all the specified Strings.

Signature

```java
public static String getCommonPrefix(List<String> strings)
```

Parameters

`strings`
Type: `List<String>`

Return Value

Type: `String`

Example

```java
List<String> ls = new List<String>{'SFDCApex', 'SFDCVisualforce'};
String prefix = String.getCommonPrefix(ls);
System.assertEquals('SFDC', prefix);
```
getLevenshteinDistance(stringToCompare)

Returns the Levenshtein distance between the current String and the specified String.

Signature

```java
public Integer getLevenshteinDistance(String stringToCompare)
```

Parameters

- `stringToCompare`
  Type: `String`

Return Value

Type: `Integer`

Usage

The Levenshtein distance is the number of changes needed to change one String into another. Each change is a single character modification (deletion, insertion or substitution).

Example

```java
String s = 'Hello Joe';
Integer i = s.getLevenshteinDistance('Hello Max');
System.assertEquals(3, i);
```

getLevenshteinDistance(stringToCompare, threshold)

Returns the Levenshtein distance between the current String and the specified String if it is less than or equal than the given threshold; otherwise, returns -1.

Signature

```java
public Integer getLevenshteinDistance(String stringToCompare, Integer threshold)
```

Parameters

- `stringToCompare`
  Type: `String`
- `threshold`
  Type: `Integer`

Return Value

Type: `Integer`
Usage

The Levenshtein distance is the number of changes needed to change one String into another. Each change is a single character modification (deletion, insertion or substitution).

Example:

In this example, the Levenshtein distance is 3, but the threshold argument is 2, which is less than the distance, so this method returns -1.

Example

```java
String s = 'Hello Jane';
Integer i = s.getLevenshteinDistance('Hello Max', 2);
System.assertEquals(-1, i);
```

hashCode()

Returns a hash code value for this string.

Signature

```java
public Integer hashCode()
```

Return Value

Type: Integer

Usage

This value is based on the hash code computed by the Java `String.hashCode` counterpart method.

You can use this method to simplify the computation of a hash code for a custom type that contains String member variables. You can compute your type's hash code value based on the hash code of each String variable. For example:

For more details about the use of hash code methods with custom types, see Using Custom Types in Map Keys and Sets.

Example

```java
public class MyCustomClass {
    String x, y;
    // Provide a custom hash code
    public Integer hashCode() {
        return (31*x.hashCode())^y.hashCode();
    }
}
```

indexOf(substring)

Returns the index of the first occurrence of the specified substring. If the substring does not occur, this method returns -1.
Signature

public Integer indexOf(String substring)

Parameters

substring
  Type: String

Return Value

Type: Integer

Example

String myString1 = 'abcde';
String myString2 = 'cd';
Integer result = myString1.indexOf(myString2);
System.assertEquals(2, result);

indexOf(substring, index)

Returns the zero-based index of the first occurrence of the specified substring from the point of the given index. If the substring does not occur, this method returns -1.

Signature

public Integer indexOf(String substring, Integer index)

Parameters

substring
  Type: String

index
  Type: Integer

Return Value

Type: Integer

Example

String myString1 = 'abcdabcd';
String myString2 = 'ab';
Integer result = myString1.indexOf(myString2, 1);
System.assertEquals(4, result);

indexOfAny(substring)

Returns the zero-based index of the first occurrence of any character specified in the substring. If none of the characters occur, returns -1.

Signature

public Integer indexOfAny(String substring)
Signature

downloads\.indexOfAny(String substring)

Parameters

substring
  Type: String

Return Value

Type: Integer

Example

String s1 = 'abcd';
String s2 = 'xc';
Integer result = s1.indexOfAny(s2);
System.assertEquals(2, result);

indexOfAnyBut(substring)

Returns the zero-based index of the first occurrence of a character that is not in the specified substring. Otherwise, returns -1.

Signature

public Integer indexOfAnyBut(String substring)

Parameters

substring
  Type: String

Return Value

Type: Integer

Example

String s1 = 'abcd';
String s2 = 'xc';
Integer result = s1.indexOfAnyBut(s2);
System.assertEquals(0, result);

indexOfChar(character)

Returns the index of the first occurrence of the character that corresponds to the specified character value.

Signature

public Integer indexOfChar(Integer character)
Parameters

character
Type: Integer
The integer value of the character in the string.

Return Value

Type: Integer
The index of the first occurrence of the specified character, -1 if the character is not found.

Usage

The index that this method returns is in Unicode code units.

Example

```java
String str = '\u03A9 is Ω (Omega)';
// Returns 0, which is the first character.
System.debug('indexOfChar(937)=' + str.indexOfChar(937));
// Output:
// indexOfChar(937)=0
```

`indexOfChar(character, startIndex)`

Returns the index of the first occurrence of the character that corresponds to the specified character value, starting from the specified index.

Signature

```
public Integer indexOfChar(Integer character, Integer startIndex)
```

Parameters

character
Type: Integer
The integer value of the character to look for.

startIndex
Type: Integer
The index to start the search from.

Return Value

Type: Integer
The index, starting from the specified start index, of the first occurrence of the specified character, -1 if the character is not found.
Usage
The index that this method returns is in Unicode code units.

Example
This example shows different ways of searching for the index of the Omega character. The first call to `indexOfChar` doesn’t specify a start index and therefore the returned index is 0, which is the first occurrence of Omega in the entire string. The subsequent calls specify a start index to find the occurrence of Omega in substrings that start at the specified index.

```java
String str = 'Ω and \u03A9 and Ω';
System.debug('indexOfChar(937)= ' + str.indexOfChar(937));
System.debug('indexOfChar(937,1)= ' + str.indexOfChar(937,1));
System.debug('indexOfChar(937,10)= ' + str.indexOfChar(937,10));
```

// Output:
// indexOfChar(937)=0
// indexOfChar(937,1)=6, (corresponds to the escaped form \u03A9)
// indexOfChar(937,10)=12

`indexOfDifference(stringToCompare)`
Returns the zero-based index of the character where the current String begins to differ from the specified String.

Signature
```
public Integer indexOfDifference(String stringToCompare)
```

Parameters

- `stringToCompare`  
  Type: String

Return Value

Type: Integer

Example

```java
String s1 = 'abcd';
String s2 = 'abxc';
Integer result = s1.indexOfDifference(s2);
System.assertEquals(2, result);
```

`indexOfIgnoreCase(substring)`
Returns the zero-based index of the first occurrence of the specified substring without regard to case. If the substring does not occur, this method returns -1.

Signature
```
public Integer indexOfIgnoreCase(String substring)
```

3435
Parameters

substring
Type: String

Return Value
Type: Integer

Example

```java
String s1 = 'abcd';
String s2 = 'BC';
Integer result = s1.indexOfIgnoreCase(s2, 0);
System.assertEquals(1, result);
```

`indexOfIgnoreCase(substring, startPosition)`

Returns the zero-based index of the first occurrence of the specified substring from the point of index `i`, without regard to case. If the substring does not occur, this method returns -1.

Signature

```java
public Integer indexOfIgnoreCase(String substring, Integer startPosition)
```

Parameters

substring
Type: String

startPosition
Type: Integer

Return Value
Type: Integer

`isAllLowerCase()`

Returns `true` if all characters in the current String are lowercase; otherwise, returns `false`.

Signature

```java
public Boolean isAllLowerCase()
```

Return Value
Type: Boolean
**Example**

```java
String allLower = 'abcde';
System.assert(allLower.isAllLowerCase());
```

**isAllUpperCase()**

Returns `true` if all characters in the current String are uppercase; otherwise, returns `false`.

**Signature**

```java
public Boolean isAllUpperCase()
```

**Return Value**

Type: `Boolean`

**Example**

```java
String allUpper = 'ABCDE';
System.assert(allUpper.isAllUpperCase());
```

**isAlpha()**

Returns `true` if all characters in the current String are Unicode letters only; otherwise, returns `false`.

**Signature**

```java
public Boolean isAlpha()
```

**Return Value**

Type: `Boolean`

**Example**

```java
// Letters only
String s1 = 'abc';
// Returns true
Boolean b1 =
    s1.isAlpha();
System.assertEquals(
    true, b1);

// Letters and numbers
String s2 = 'abc 21';
// Returns false
Boolean b2 =
    s2.isAlpha();
System.assertEquals(
    false, b2);
```
**isAlphaSpace()**

Returns `true` if all characters in the current String are Unicode letters or spaces only; otherwise, returns `false`.

**Signature**

```java
public Boolean isAlphaSpace()
```

**Return Value**

Type: `Boolean`

**Example**

```java
String alphaSpace = 'aA Bb';
System.assert(alphaSpace.isAlphaSpace());
String notAlphaSpace = 'ab 12';
System.assert(!notAlphaSpace.isAlphaSpace());
notAlphaSpace = 'aA$Bb';
System.assert(!notAlphaSpace.isAlphaSpace());
```

**isAlphanumeric()**

Returns `true` if all characters in the current String are Unicode letters or numbers only; otherwise, returns `false`.

**Signature**

```java
public Boolean isAlphanumeric()
```

**Return Value**

Type: `Boolean`

**Example**

```java
// Letters only
String s1 = 'abc';
// Returns true
Boolean b1 = s1.isAlphanumeric();
System.assertEquals(true, b1);
// Letters and numbers
String s2 = 'abc021';
// Returns true
Boolean b2 = s2.isAlphanumeric();
System.assertEquals(true, b2);
```
**isAlphanumericSpace()**

Returns **true** if all characters in the current String are Unicode letters, numbers, or spaces only; otherwise, returns **false**.

**Signature**

```java
public Boolean isAlphanumericSpace()
```

**Return Value**

Type: Boolean

**Example**

```java
String alphanumSpace = 'AE 86';
System.assert(alphanumSpace.isAlphanumericSpace());
String notAlphanumSpace = 'aA$12';
System.assert(!notAlphanumSpace.isAlphanumericSpace());
```

**isAsciiPrintable()**

Returns **true** if the current String contains only ASCII printable characters; otherwise, returns **false**.

**Signature**

```java
public Boolean isAsciiPrintable()
```

**Return Value**

Type: Boolean

**Example**

```java
String ascii = 'abcd1234!#$^&*()~--+={[]}|:<,.?';
System.assert(ascii.isAsciiPrintable());
String notAscii = '√';
System.assert(!notAscii.isAsciiPrintable());
```

**isBlank(inputString)**

Returns **true** if the specified String is white space, empty (""), or null; otherwise, returns **false**.

**Signature**

```java
public static Boolean isBlank(String inputString)
```

**Parameters**

`inputString`

Type: String
Return Value
Type: Boolean

Example
```
String blank = '';
String nullString = null;
String whitespace = ' ';
System.assert(String.isBlank(blank));
System.assert(String.isBlank(nullString));
System.assert(String.isBlank(whitespace));
String alpha = 'Hello';
System.assert(!String.isBlank(alpha));
```

isEmpty(inputString)
Returns true if the specified String is empty ("") or null; otherwise, returns false.

Signature
```
public static Boolean isEmpty(String inputString)
```

Parameters
- inputString
  Type: String

Return Value
Type: Boolean

Example
```
String empty = '';
String nullString = null;
System.assert(String.isEmpty(empty));
System.assert(String.isEmpty(nullString));
String whitespace = ' ';
String alpha = 'Hello';
System.assert(!String.isEmpty(whitespace));
System.assert(!String.isEmpty(alpha));
```

isNotBlank(inputString)
Returns true if the specified String is not whitespace, not empty (""), and not null; otherwise, returns false.

Signature
```
public static Boolean isNotBlank(String inputString)
```

Example
```
```
Parameters

inputString
Type: String

Return Value

Type: Boolean

Example

```java
String alpha = 'Hello world!';
System.assert(String.isNotBlank(alpha));
String blank = '';
String nullString = null;
String whitespace = ' ';
System.assert(!String.isNotBlank(blank));
System.assert(!String.isNotBlank(nullString));
System.assert(!String.isNotBlank(whitespace));
```

isEmpty(inputString)

Returns true if the specified String is not empty ("") and not null; otherwise, returns false.

Signature

```java
public static Boolean isEmpty(String inputString)
```

Parameters

inputString
Type: String

Return Value

Type: Boolean

Example

```java
String whitespace = ' ';
String alpha = 'Hello world!';
System.assert(String.isNotEmpty(whitespace));
System.assert(String.isNotEmpty(alpha));
String empty = '';  
String nullString = null;
System.assert(!String.isNotEmpty(empty));
System.assert(!String.isNotEmpty(nullString));
```

isNumeric()

Returns true if the current String contains only Unicode digits; otherwise, returns false.
Signature

public Boolean isNumeric()

Return Value

Type: Boolean

Usage

A decimal point (1.2) is not a Unicode digit.

Example

```java
String numeric = '1234567890';
System.assert(numeric.isNumeric());
String alphanumeric = 'R32';
String decimalPoint = '1.2';
System.assert(!alphanumeric.isNumeric());
System.assert(!decimalPoint.isNumeric());
```

isNumericSpace()

Returns true if the current String contains only Unicode digits or spaces; otherwise, returns false.

Signature

public Boolean isNumericSpace()

Return Value

Type: Boolean

Usage

A decimal point (1.2) is not a Unicode digit.

Example

```java
String numericSpace = '1 2 3';
System.assert(numericSpace.isNumericSpace());
String notNumericspace = 'FD3S FC3S';
System.assert(!notNumericspace.isNumericspace());
```

isWhitespace()

Returns true if the current String contains only white space characters or is empty; otherwise, returns false.

Signature

public Boolean isWhitespace()
Return Value
Type: Boolean

Example
```java
String whitespace = ' ';
String blank = '';  
System.assert(whitespace.isWhitespace());
System.assert(blank.isWhitespace());
String alphanum = 'SIL80';
System.assert(!alphanum.isWhitespace());
```

**join(iterableObj, separator)**
Joins the elements of the specified iterable object, such as a List, into a single String separated by the specified separator.

**Signature**
```java
public static String join(Object iterableObj, String separator)
```

**Parameters**
- **iterableObj**
  Type: Object
- **separator**
  Type: String

**Return Value**
Type: String

**Usage**
```java
List<Integer> li = new List<Integer>{10, 20, 30};
String s = String.join(li, '/');
System.assertEquals('10/20/30', s);
```

**lastIndexOf(substring)**
Returns the index of the last occurrence of the specified substring. If the substring does not occur, this method returns -1.

**Signature**
```java
public Integer lastIndexOf(String substring)
```
Parameters

substring
Type: String

Return Value
Type: Integer

Example

```java
String s1 = 'abcdefgc';
Integer i1 = s1.lastIndexOf('c');
System.assertEquals(7, i1);
```

`lastIndexOf(substring, endPosition)`

Returns the index of the last occurrence of the specified substring, starting from the character at index 0 and ending at the specified index.

Signature

```java
public Integer lastIndexOf(String substring, Integer endPosition)
```

Parameters

substring
Type: String

definition

Type: Integer

Usage

If the substring doesn't occur or `endPosition` is negative, this method returns -1. If `endPosition` is larger than the last index in the current String, the entire String is searched.

Example

```java
String s1 = 'abcdaacd';
Integer i1 = s1.lastIndexOf('c', 7);
System.assertEquals(6, i1);
Integer i2 = s1.lastIndexOf('c', 3);
System.assertEquals(2, i2);
```

`lastIndexOfChar(character)`

Returns the index of the last occurrence of the character that corresponds to the specified character value.
Signature
public Integer lastIndexOfChar(Integer character)

Parameters
character
Type: Integer
The integer value of the character in the string.

Return Value
Type: Integer
The index of the last occurrence of the specified character, -1 if the character is not found.

Usage
The index that this method returns is in Unicode code units.

Example
String str = '\u03A9 is Ω (Omega)';
// Get the last occurrence of Omega.
System.assertEquals(5, str.lastIndexOfChar(937));

lastIndexOfChar(character, endIndex)
Returns the index of the last occurrence of the character that corresponds to the specified character value, starting from the specified index.

Signature
public Integer lastIndexOfChar(Integer character, Integer endIndex)

Parameters
character
Type: Integer
The integer value of the character to look for.
endIndex
Type: Integer
The index to end the search at.

Return Value
Type: Integer
The index, starting from the specified start index, of the last occurrence of the specified character. -1 if the character is not found.
Usage

The index that this method returns is in Unicode code units.

Example

This example shows different ways of searching for the index of the last occurrence of the Omega character. The first call to `lastIndexOfChar` doesn’t specify an end index and therefore the returned index is 12, which is the last occurrence of Omega in the entire string. The subsequent calls specify an end index to find the last occurrence of Omega in substrings.

```java
String str = 'Ω and \u03A9 and Ω';
System.assertEquals(12, str.lastIndexOfChar(937));
System.assertEquals(6, str.lastIndexOfChar(937,11));
System.assertEquals(0, str.lastIndexOfChar(937,5));
```

`lastIndexOfIgnoreCase(substring)`

Returns the index of the last occurrence of the specified substring regardless of case.

Signature

```java
public Integer lastIndexOfIgnoreCase(String substring)
```

Parameters

`substring`
Type: `String`

Return Value

Type: `Integer`

Usage

If the substring doesn’t occur, this method returns -1.

Example

```java
String s1 = 'abcdaacd';
Integer i1 = s1.lastIndexOfIgnoreCase('DAAC');
System.assertEquals(3, i1);
```

`lastIndexOfIgnoreCase(substring, endPosition)`

Returns the index of the last occurrence of the specified substring regardless of case, starting from the character at index 0 and ending at the specified index.

Signature

```java
public Integer lastIndexOfIgnoreCase(String substring, Integer endPosition)
```
Parameters

substring
Type: String

denPosition
Type: Integer

Return Value
Type: Integer

Usage
If the substring doesn't occur or endPosition is negative, this method returns -1. If endPosition is larger than the last index in the current String, the entire String is searched.

Example

```java
String s1 = 'abcdaacd';
Integer i1 = s1.lastIndexOfIgnoreCase('C', 7);
System.assertEquals(6, i1);
```

left(length)

Returns the leftmost characters of the current String of the specified length.

Signature

```java
public String left(Integer length)
```

Parameters

length
Type: Integer

Return Value
Type: String

Usage
If length is greater than the String size, the entire String is returned.

Example

```java
String s1 = 'abcdaacd';
String s2 = s1.left(3);
System.assertEquals('abc', s2);
```
**leftPad(length)**

Returns the current String padded with spaces on the left and of the specified length.

**Signature**

```java
public String leftPad(Integer length)
```

**Parameters**

- `length`
  
  Type: `Integer`

**Usage**

If `length` is less than or equal to the current String size, the entire String is returned without space padding.

**Return Value**

Type: `String`

**Example**

```java
String s1 = 'abc';
String s2 = s1.leftPad(5);
System.assertEquals(' abc', s2);
```

**leftPad(length, padStr)**

Returns the current String padded with String `padStr` on the left and of the specified length.

**Signature**

```java
public String leftPad(Integer length, String padStr)
```

**Parameters**

- `length`
  
  Type: `Integer`

- `padStr`
  
  Type: `String`

  String to pad with; if null or empty treated as single blank.

**Usage**

If `length` is less than or equal to the current String size, the entire String is returned without space padding.

**Return Value**

Type: `String`
Example

```java
String s1 = 'abc';
String s2 = 'xy';
String s3 = s1.leftPad(7, s2);
System.assertEquals('xyxyabc', s3);
```

length()

Returns the number of 16-bit Unicode characters contained in the String.

Signature

```java
public Integer length()
```

Return Value

Type: Integer

Example

```java
String myString = 'abcd';
Integer result = myString.length();
System.assertEquals(result, 4);
```

mid(startIndex, length)

Returns a new String that begins with the character at the specified zero-based `startIndex` with the number of characters specified by `length`.

Signature

```java
public String mid(Integer startIndex, Integer length)
```

Parameters

- `startIndex`
  - Type: `Integer`
  - If `startIndex` is negative, it is considered to be zero.

- `length`
  - Type: `Integer`
  - If `length` is negative or zero, an empty String is returned. If `length` is greater than the remaining characters, the remainder of the String is returned.

Return Value

Type: `String`
Usage

This method is similar to the `substring(startIndex)` and `substring(startIndex, endIndex)` methods, except that the second argument is the number of characters to return.

Example

```java
String s = 'abcde';
String s2 = s.mid(2, 3);
System.assertEquals('cde', s2);
```

**normalizeSpace()**

Returns the current String with leading, trailing, and repeating white space characters removed.

**Signature**

```java
public String normalizeSpace()
```

**Return Value**

Type: String

**Usage**

This method normalizes the following white space characters: space, tab (\t), new line (\n), carriage return (\r), and form feed (\f).

**Example**

```java
String s1 = 'Salesforce \t force.com';
String s2 = s1.normalizeSpace();
System.assertEquals('Salesforce force.com', s2);
```

**offsetByCodePoints(index, codePointOffset)**

Returns the index of the Unicode code point that is offset by the specified number of code points, starting from the given index.

**Signature**

```java
public Integer offsetByCodePoints(Integer index, Integer codePointOffset)
```

**Parameters**

- `index`  
  Type: Integer  
  The start index in the string.
**codePointOffset**

Type: `Integer`

The number of code points to be offset.

**Return Value**

Type: `Integer`

The index that corresponds to the start index that is added to the offset.

**Usage**

Unpaired surrogates within the text range that is specified by `index` and `codePointOffset` count as one code point each.

**Example**

This example calls `offsetByCodePoints` on a string with a start index of 0 (to start from the first character) and an offset of three code points. The string contains one sequence of supplementary characters in escaped form (a pair of characters). After an offset of three code points when counting from the beginning of the string, the returned code point index is four.

```java
String str = 'A \uD835\uDD0A BC';
System.assertEquals(4, str.offsetByCodePoints(0,3));
```

**remove(substring)**

Removes all occurrences of the specified substring and returns the String result.

**Signature**

```java
public String remove(String substring)
```

**Parameters**

- `substring`
  
  Type: `String`

**Return Value**

Type: `String`

**Example**

```java
String s1 = 'Salesforce and force.com';
String s2 =
  s1.remove('force');
System.assertEquals(
  'Sales and .com', s2);
```

**removeEnd(substring)**

Removes the specified substring only if it occurs at the end of the String.
Signature

public String removeEnd(String substring)

Parameters

substring
Type: String

Return Value
Type: String

Example

```java
String s1 = 'Salesforce and force.com';
String s2 =
    s1.removeEnd('.com');
System.assertEquals(
    'Salesforce and force', s2);
```

removeEndIgnoreCase(substring)
Removes the specified substring only if it occurs at the end of the String using a case-insensitive match.

Signature

public String removeEndIgnoreCase(String substring)

Parameters

substring
Type: String

Return Value
Type: String

Example

```java
String s1 = 'Salesforce and force.com';
String s2 = s1.removeEndIgnoreCase('.COM');
System.assertEquals('Salesforce and force', s2);
```

removeStart(substring)
Removes the specified substring only if it occurs at the beginning of the String.

Signature

public String removeStart(String substring)
Parameters

substring
Type: String

Return Value
Type: String

Example

```java
String s1 = 'Salesforce and force.com';
String s2 =
    s1.removeStart('Sales');
System.assertEquals(
    'force and force.com', s2);
```

**removeStartIgnoreCase(substring)**

Removes the specified substring only if it occurs at the beginning of the String using a case-insensitive match.

Signature

```java
public String removeStartIgnoreCase(String substring)
```

Parameters

substring
Type: String

Return Value
Type: String

Example

```java
String s1 = 'Salesforce and force.com';
String s2 =
    s1.removeStartIgnoreCase('SALES');
System.assertEquals(
    'force and force.com', s2);
```

**repeat(numberOfTimes)**

Returns the current String repeated the specified number of times.

Signature

```java
public String repeat(Integer numberOfTimes)
```
Parameters

numberOfTimes
Type: Integer

Return Value
Type: String

Example

```java
String s1 = 'SFDC';
String s2 = s1.repeat(2);
System.assertEquals('SFDCSFDC', s2);
```

**repeat(separator, numberOfTimes)**

Returns the current String repeated the specified number of times using the specified separator to separate the repeated Strings.

**Signature**

```java
public String repeat(String separator, Integer numberOfTimes)
```

**Parameters**

separator
Type: String

numberOfTimes
Type: Integer

**Return Value**
Type: String

**Example**

```java
String s1 = 'SFDC';
String s2 = s1.repeat('-', 2);
System.assertEquals('SFDC-SFDC', s2);
```

**replace(target, replacement)**

Replaces each substring of a string that matches the literal target sequence `target` with the specified literal replacement sequence `replacement`.

**Signature**

```java
public String replace(String target, String replacement)
```
Parameters

target
  Type: String

replacement
  Type: String

Return Value
  Type: String

Example

```java
String s1 = 'abcdbca';
String target = 'bc';
String replacement = 'xy';
String s2 = s1.replace(target, replacement);
System.assertEquals('axydxya', s2);
```

`replaceAll(regExp, replacement)`

Replaces each substring of a string that matches the regular expression `regExp` with the replacement sequence `replacement`.

Signature

```java
public String replaceAll(String regExp, String replacement)
```

Parameters

`regExp`
  Type: String

`replacement`
  Type: String

Return Value
  Type: String

Usage

See the Java `Pattern` class for information on regular expressions.

Example

```java
String s1 = 'a b c 5 xyz';
String regExp = '[a-zA-Z]';
String replacement = '1';
String s2 = s1.replaceAll(regExp, replacement);
System.assertEquals('1 1 1 5 111', s2);
```
**replaceFirst(regExp, replacement)**

Replaces the first substring of a string that matches the regular expression `regExp` with the replacement sequence `replacement`.

**Signature**

```java
public String replaceFirst(String regExp, String replacement)
```

**Parameters**

- `regExp`
  - Type: `String`
- `replacement`
  - Type: `String`

**Return Value**

Type: `String`

**Usage**

See the Java `Pattern` class for information on regular expressions.

**Example**

```java
String s1 = 'a b c 11 xyz';
String regExp = '[a-zA-Z]{2}';
String replacement = '2';
String s2 = s1.replaceFirst(regExp, replacement);
System.assertEquals('a b c 11 2z', s2);
```

**reverse()**

Returns a String with all the characters reversed.

**Signature**

```java
public String reverse()
```

**Return Value**

Type: `String`

**right(length)**

Returns the rightmost characters of the current String of the specified length.

**Signature**

```java
public String right(Integer length)
```
Parameters

\( \text{length} \)
Type: Integer

If \( \text{length} \) is greater than the String size, the entire String is returned.

Return Value
Type: String

Example

```java
String s1 = 'Hello Max';
String s2 =  
s1.right(3);
System.assertEquals(
 'Max', s2);
```

**rightPad(length)**

Returns the current String padded with spaces on the right and of the specified length.

**Signature**

```
public String rightPad(Integer length)
```

Parameters

\( \text{length} \)
Type: Integer

If \( \text{length} \) is less than or equal to the current String size, the entire String is returned without space padding.

Return Value
Type: String

Example

```java
String s1 = 'abc';
String s2 =  
s1.rightPad(5);
System.assertEquals(
 'abc ', s2);
```

**rightPad(length, padStr)**

Returns the current String padded with String \( \text{padStr} \) on the right and of the specified length.
Signature

public String rightPad(Integer length, String padStr)

Parameters

length
Type: Integer

padStr
Type: String
String to pad with; if null or empty treated as single blank.

Usage

If length is less than or equal to the current String size, the entire String is returned without space padding.

Return Value

Type: String

Example

```java
String s1 = 'abc';
String s2 = 'xy';
String s3 = s1.rightPad(7, s2);
System.assertEquals('abcxyxy', s3);
```

split(regExp)

Returns a list that contains each substring of the String that is terminated by either the regular expression regExp or the end of the String.

Signature

public String[] split(String regExp)

Parameters

regExp
Type: String

Return Value

Type: String[]

Note: In API version 34.0 and earlier, a zero-width regExp value produces an empty list item at the beginning of the method's output.

Usage

See the Java Pattern class for information on regular expressions.
The substrings are placed in the list in the order in which they occur in the String. If \textit{regExp} does not match any part of the String, the resulting list has just one element containing the original String.

\textbf{Example}

In the following example, a string is split using a backslash as a delimiter.

\begin{verbatim}
public String splitPath(String filename) {
    if (filename == null)
        return null;
    List<String> parts = filename.split("\\");
    filename = parts[parts.size()-1];
    return filename;
}
\end{verbatim}

// For example, if the file path is e:\\processed\\PPDSF100111.csv
// This method splits the path and returns the last part.
// Returned filename is PPDSF100111.csv

\textbf{split}(\textit{regExp}, \textit{limit})

Returns a list that contains each substring of the String that is terminated by either the regular expression \textit{regExp} or the end of the String.

\textbf{Signature}

\begin{verbatim}
public String[] split(String regExp, Integer limit)
\end{verbatim}

\textbf{Parameters}

\begin{itemize}
    \item \textit{regExp} \\
        Type: \textit{String} \\
        A regular expression.
    \item \textit{limit} \\
        Type: \textit{Integer}
\end{itemize}

\textbf{Return Value}

Type: \textit{String[]}

\begin{itemize}
    \item \textbf{Note:} In API version 34.0 and earlier, a zero-width \textit{regExp} value produces an empty list item at the beginning of the method’s output.
\end{itemize}

\textbf{Usage}

The optional \textit{limit} parameter controls the number of times the pattern is applied and therefore affects the length of the list.

- If \textit{limit} is greater than zero:
  - The pattern is applied a maximum of \((\textit{limit} - 1)\) times.
  - The list’s length is no greater than \textit{limit}.
  - The list’s last entry contains all input beyond the last matched delimiter.
• If \( \text{limit} \) is non-positive, the pattern is applied as many times as possible, and the list can have any length.
• If \( \text{limit} \) is zero, the pattern is applied as many times as possible, the list can have any length, and trailing empty strings are discarded.

Example
For example, for `String s = 'boo:and:moo'`:

- `s.split(';', 2)` results in `{'boo', 'and:moo'}`
- `s.split(';', 5)` results in `{'boo', 'and', 'moo'}`
- `s.split(';', -2)` results in `{'boo', 'and', 'moo'}`
- `s.split('o', 5)` results in `{'b', '', ':and:m', '', ''}`
- `s.split('o', -2)` results in `{'b', '', ':and:m', '', ''}`
- `s.split('o', 0)` results in `{'b', '', ':and:m'}`

splitByCharacterType()
Splits the current String by character type and returns a list of contiguous character groups of the same type as complete tokens.

Signature
```
public List<String> splitByCharacterType()
```

Return Value
Type: `List<String>`

Usage
For more information about the character types used, see `java.lang.Character.getType(char)`.

Example
```
String s1 = 'Lightning.platform';
List<String> ls =
    s1.splitByCharacterType();
System.debug(ls);
// Writes this output:
// (L, ightning, ., platform)
```

splitByCharacterTypeCamelCase()
Splits the current String by character type and returns a list of contiguous character groups of the same type as complete tokens, with the following exception: the uppercase character, if any, immediately preceding a lowercase character token belongs to the following character token rather than to the preceding.

Signature
```
public List<String> splitByCharacterTypeCamelCase()
```
Return Value
Type: List<String>

Usage
For more information about the character types used, see java.lang.Character.getType(char).

Example
```java
String s1 = 'Lightning.platform';
List<String> ls =
    s1.splitByCharacterTypeCamelCase();
System.debug(ls);
// Writes this output:
// (Lightning, ., platform)
```

**startsWith (prefix)**

Returns true if the String that called the method begins with the specified prefix.

**Signature**
```
public Boolean startsWith(String prefix)
```

**Parameters**
```
prefix
  Type: String
```

Return Value
Type: Boolean

**Example**
```java
String s1 = 'AE86 vs EK9';
System.assert(s1.startsWith('AE86'));
```

**startsWithIgnoreCase (prefix)**

Returns true if the current String begins with the specified prefix regardless of the prefix case.

**Signature**
```
public Boolean startsWithIgnoreCase(String prefix)
```

**Parameters**
```
prefix
  Type: String
```

**Example**
```java
String s1 = 'AE86 vs EK9';
System.assert(s1.startsWithIgnoreCase('AE86'));
```
Return Value
Type: Boolean

Example

```java
String s1 = 'AE86 vs EK9';
System.assert(s1.startsWithIgnoreCase('ae86'));
```

stripHtmlTags()
Removes HTML markup and returns plain text.

Signature

```java
public String stripHtmlTags(String htmlInput)
```

Return Value
Type: String

Usage

⚠️ **Warning:** The stripHtmlTags function does not recursively strip tags; therefore, tags may still exist in the returned string. Do not use the stripHtmlTags function to sanitize input for inclusion as a raw HTML page. The unescaped output is not considered safe to include in an HTML document. The function will be deprecated in a future release.

Example

```java
String s1 = '<b>hello world</b>';  
String s2 = s1.stripHtmlTags();  
System.assertEquals('hello world', s2);
```

substring(startIndex)
Returns a new String that begins with the character at the specified zero-based startIndex and extends to the end of the String.

Signature

```java
public String substring(Integer startIndex)
```

Parameters

startIndex
Type: Integer

Return Value
Type: String
Example

```java
String s1 = 'hamburger';
System.assertEquals('burger', s1.substring(3));
```

**substring**(startIndex, endIndex)

Returns a new String that begins with the character at the specified zero-based `startIndex` and extends to the character at `endIndex` - 1.

**Signature**

```java
public String substring(Integer startIndex, Integer endIndex)
```

**Parameters**

- **startIndex**
  - Type: Integer
- **endIndex**
  - Type: Integer

**Return Value**

Type: String

**Example**

```java
'hamburger'.substring(4, 8);
// Returns "urge"

'smiles'.substring(1, 5);
// Returns "mile"
```

**substringAfter**(separator)

Returns the substring that occurs after the first occurrence of the specified separator.

**Signature**

```java
public String substringAfter(String separator)
```

**Parameters**

- **separator**
  - Type: String

**Return Value**

Type: String
Example

```java
String s1 = 'Salesforce.Lightning.platform';
String s2 =
    s1.substringAfter('.');
System.assertEquals('Lightning.platform', s2);
```

`substringAfterLast(separator)`

Returns the substring that occurs after the last occurrence of the specified separator.

Signature

```java
public String substringAfterLast(String separator)
```

Parameters

`separator`

Type: `String`

Return Value

Type: `String`

Example

```java
String s1 = 'Salesforce.Lightning.platform';
String s2 =
    s1.substringAfterLast('.');
System.assertEquals('platform', s2);
```

`substringBefore(separator)`

Returns the substring that occurs before the first occurrence of the specified separator.

Signature

```java
public String substringBefore(String separator)
```

Parameters

`separator`

Type: `String`

Return Value

Type: `String`
Example

```java
String s1 = 'Salesforce.Lightning.platform';
String s2 =
    s1.substringBefore('.');
System.assertEquals('Salesforce', s2);
```

**substringBeforeLast(separator)**

Returns the substring that occurs before the last occurrence of the specified separator.

**Signature**

```java
public String substringBeforeLast(String separator)
```

**Parameters**

- `separator`
  - Type: `String`

**Return Value**

- Type: `String`

Example

```java
String s1 = 'Salesforce.Lightning.platform';
String s2 =
    s1.substringBeforeLast('.');
System.assertEquals('Salesforce.Lightning', s2);
```

**substringBetween(tag)**

Returns the substring that occurs between two instances of the specified `tag` `String`.

**Signature**

```java
public String substringBetween(String tag)
```

**Parameters**

- `tag`
  - Type: `String`

**Return Value**

- Type: `String`
Example

```java
String s1 = 'tagYellowtag';
String s2 = s1.substringBetween('tag');
System.assertEquals('Yellow', s2);
```

`substringBetween(open, close)`

Returns the substring that occurs between the two specified Strings.

Signature

```java
public String substringBetween(String open, String close)
```

Parameters

- `open`
  - Type: String

- `close`
  - Type: String

Return Value

Type: String

Example

```java
String s1 = 'xYellowy';
String s2 =
    s1.substringBetween('x','y');
System.assertEquals('Yellow', s2);
```

`swapCase()`

Swaps the case of all characters and returns the resulting String by using the default (English US) locale.

Signature

```java
public String swapCase()
```

Return Value

Type: String

Usage

Upper case and title case converts to lower case, and lower case converts to upper case.
Example

```java
String s1 = 'Force.com';
String s2 = s1.swapCase();
System.assertEquals('fORCE.COM', s2);
```

toLowerCase()

Converts all of the characters in the String to lowercase using the rules of the default (English US) locale.

**Signature**

```java
public String toLowerCase()
```

**Return Value**

Type: `String`

**Example**

```java
String s1 = 'This is hard to read';
System.assertEquals('this is hard to read',
                   s1.toLowerCase());
```

toLowerCase(locale)

Converts all of the characters in the String to lowercase using the rules of the specified locale.

**Signature**

```java
public String toLowerCase(String locale)
```

**Parameters**

- `locale`
  Type: `String`

**Return Value**

Type: `String`

**Example**

```java
// Example in Turkish
// An uppercase dotted "i", \u0304, which is İ
// Note this contains both a İ as well as a I
String s1 = 'KIYMETLİ';
String s1Lower = s1.toLowerCase('tr');
// Dotless lowercase "i", \u0131, which is ı
// Note this has both a i and ı
String expected = 'kıymetli';
```
System.assertEquals(expected, s1Lower);
// Note if this was done in toLowerCase('en'), it would output 'kiymetli'

toUpperCase()

Converts all of the characters in the String to uppercase using the rules of the default (English US) locale.

**Signature**

```java
public String toUpperCase()
```

**Return Value**

Type: String

**Example**

```java
String myString1 = 'abcd';
String myString2 = 'ABCD';
myString1 =
    myString1.toUpperCase();
Boolean result =
    myString1.equals(myString2);
System.assertEquals(result, true);
```

toUpperCase(locale)

Converts all of the characters in the String to the uppercase using the rules of the specified locale.

**Signature**

```java
public String toUpperCase(String locale)
```

**Parameters**

locale
Type: String

**Return Value**

Type: String

**Example**

```java
// Example in Turkish
// Dotless lowercase "i", \u0131, which is ı
// Note this has both a i and ı
String s1 = 'imkansız';
String s1Upper = s1.toUpperCase('tr');
// An uppercase dotted "i", \u0304, which is İ
// Note this contains both a İ as well as a I
```
trim()

Returns a copy of the string that no longer contains any leading or trailing white space characters.

Signature

public String trim()

Return Value

Type: String

Usage

Leading and trailing ASCII control characters such as tabs and newline characters are also removed. White space and control characters that aren’t at the beginning or end of the sentence aren’t removed.

Example

String s1 = ' Hello! ';
String trimmed = s1.trim();
System.assertEquals('Hello!', trimmed);

uncapitalize()

Returns the current String with the first letter in lowercase.

Signature

public String uncapitalize()

Return Value

Type: String

Example

String s1 = 'Hello max';
String s2 = s1.uncapitalize();
System.assertEquals('hello max', s2);
unescapeCsv()

Returns a String representing an unescaped CSV column.

Signature

public String unescapeCsv()

Return Value

Type: String

Usage

If the String is enclosed in double quotes and contains a comma, newline or double quote, quotes are removed. Also, any double quote escaped characters (a pair of double quotes) are unescaped to just one double quote.

If the String is not enclosed in double quotes, or is and does not contain a comma, newline or double quote, it is returned unchanged.

Example

```java
String s1 = "Max1, "Max2"");
String s2 = s1.unescapeCsv();
System.assertEquals('Max1, "Max2"', s2);
```

unescapeEcmaScript()

Unescapes any EcmaScript literals found in the String.

Signature

public String unescapeEcmaScript()

Return Value

Type: String

Example

```java
String s1 = "'3.8',"3.9"';
String s2 = s1.unescapeEcmaScript();
System.assertEquals('"3.8","3.9"', s2);
```
unescapeHtml3()
Unescapes the characters in a String using HTML 3.0 entities.

Signature
public String unescapeHtml3()

Return Value
Type: String

Example
String s1 = '
&quot;&lt;Black&lt;White&gt;&quot;&
String s2 =
  s1.unescapeHtml3();
System.assertEquals(""<&Black&White>"",
  s2);

unescapeHtml4()
Unescapes the characters in a String using HTML 4.0 entities.

Signature
public String unescapeHtml4()

Return Value
Type: String

Usage
If an entity isn’t recognized, it is kept as is in the returned string.

Example
String s1 = '
&quot;&lt;Black&amp;White&gt;&quot;
String s2 =
  s1.unescapeHtml4();
System.assertEquals(""<&Black&White>"",
  s2);
unescapeJava()  
Returns a String whose Java literals are unescaped. Literals unescaped include escape sequences for quotes (\") and control characters, such as tab (\t), and carriage return (\n).

Signature  
public String unescapeJava()

Return Value  
Type: String
The unescaped string.

Example  
```java
String s = 'Company: \"Salesforce.com\"';
String unescapedStr = s.unescapeJava();
System.assertEquals('Company: "Salesforce.com"', unescapedStr);
```

unescapeUnicode()  
Returns a String whose escaped Unicode characters are unescaped.

Signature  
public String unescapeUnicode()

Return Value  
Type: String
The unescaped string.

Example  
```java
String s = 'De onde você é?';
String unescapedStr = s.unescapeUnicode();
System.assertEquals('De onde você é?', unescapedStr);
```

unescapeXml()  
Unescapes the characters in a String using XML entities.

Signature  
public String unescapeXml()

Return Value  
Type: String
Usage
Supports only the five basic XML entities (gt, lt, quot, amp, apos). Does not support DTDs or external entities.

Example
```
String s1 =
    '"&quot;&amp;Black&amp;White&gt;&amp;quot;';
String s2 =
    s1.unescapeXml();
System.assertEquals(
    '"&lt;Black&amp;White&gt;"',
    s2);
```

valueOf(dateToConvert)
Returns a String that represents the specified Date in the standard "yyyy-MM-dd" format.

Signature
```
public static String valueOf(Date dateToConvert)
```

Parameters
dateToConvert
Type: Date

Return Value
Type: String

Example
```
Date myDate = Date.Today();
String sDate = String.valueOf(myDate);
```

valueOf(datetimeToConvert)
Returns a String that represents the specified Datetime in the standard "yyyy-MM-dd HH:mm:ss" format for the local time zone.

Signature
```
public static String valueOf(Datetime datetimeToConvert)
```

Parameters
datetimeToConvert
Type: Datetime
Return Value
Type: String

Example
```
DateTime dt = datetime.newInstance(1996, 6, 23);
String sDateTime = String.valueOf(dt);
System.assertEquals('1996-06-23 00:00:00', sDateTime);
```

valueOf(decimalToConvert)
Returns a String that represents the specified Decimal.

Signature
```
public static String valueOf(Decimal decimalToConvert)
```

Parameters
```
decimalToConvert
  Type: Decimal
```

Return Value
Type: String

Example
```
Decimal dec = 3.14159265;
String sDecimal = String.valueOf(dec);
System.assertEquals('3.14159265', sDecimal);
```

valueOf(doubleToConvert)
Returns a String that represents the specified Double.

Signature
```
public static String valueOf(Double doubleToConvert)
```

Parameters
```
doubleToConvert
  Type: Double
```

Return Value
Type: String
Example

```java
Double myDouble = 12.34;
String myString = String.valueOf(myDouble);
System.assertEquals('12.34', myString);
```

**valueOf(integerToConvert)**

Returns a String that represents the specified Integer.

**Signature**

```java
public static String valueOf(Integer integerToConvert)
```

**Parameters**

- `integerToConvert`  
  Type: `Integer`

**Return Value**

Type: `String`

**Example**

```java
Integer myInteger = 22;
String sInteger = String.valueOf(myInteger);
System.assertEquals('22', sInteger);
```

**valueOf(longToConvert)**

Returns a String that represents the specified Long.

**Signature**

```java
public static String valueOf(Long longToConvert)
```

**Parameters**

- `longToConvert`  
  Type: `Long`

**Return Value**

Type: `String`
Example

```java
Long myLong = 123456789;
String sLong = String.valueOf(myLong);
System.assertEquals('123456789', sLong);
```

**valueOf(toConvert)**

Returns a string representation of the specified object argument.

**Signature**

```java
public static String valueOf(Object toConvert)
```

**Parameters**

`toConvert`

Type: Object

**Return Value**

Type: String

**Usage**

If the argument is not a String, the `valueOf` method converts it into a String by calling the `toString` method on the argument, if available, or any overridden `toString` method if the argument is a user-defined type. Otherwise, if no `toString` method is available, it returns a String representation of the argument.

Example

```java
List<Integer> ls =
    new List<Integer>();
ls.add(10);
ls.add(20);
String strList =
    String.valueOf(ls);
System.assertEquals(
    '(10, 20)', strList);
```

**valueOfGmt(datetimeToConvert)**

Returns a String that represents the specified Datetime in the standard "yyyy-MM-dd HH:mm:ss" format for the GMT time zone.

**Signature**

```java
public static String valueOfGmt(Datetime datetimeToConvert)
```
Parameters

datetimeToConvert
  Type: Datetime

Return Value
  Type: String

Example

```java
// For a PST timezone:
DateTime dt = datetime.newInstance(2001, 9, 14);
String sDateTime = String.valueOfGmt(dt);
System.assertEquals('2001-09-14 07:00:00', sDateTime);
```

StubProvider Interface

StubProvider is a callback interface that you can use as part of the Apex stub API to implement a mocking framework. Use this interface with the Test.createStub() method to create stubbed Apex objects for testing.

Namespace

System

Usage

The StubProvider interface allows you to define the behavior of a stubbed Apex class. The interface specifies a single method that requires implementing: handleMethodCall(). You specify the behavior of each method of the stubbed class in the handleMethodCall() method.

In your Apex test, you create a stubbed object using the Test.createStub() method. When you invoke methods on the stubbed object, StubProvider.handleMethodCall() is called, which performs the behavior that you've specified for each method.

IN THIS SECTION:
  StubProvider Methods

SEE ALSO:
  Apex Developer Guide: Build a Mocking Framework with the Stub API
  createStub(parentType, stubProvider)

StubProvider Methods

The following are methods for StubProvider.
handleMethodCall(stubbedObject, stubbedMethodName, returnType, listOfParamTypes, listOfParamNames, listOfArgs)
Use this method to define the behavior of each method of a stubbed class.

handleMethodCall(stubbedObject, stubbedMethodName, returnType, listOfParamTypes, listOfParamNames, listOfArgs)
Use this method to define the behavior of each method of a stubbed class.

Signature

public Object handleMethodCall(Object stubbedObject, String stubbedMethodName, System.Type returnType, List<System.Type> listOfParamTypes, List<String> listOfParamNames, List<Object> listOfArgs)

Parameters

stubbedObject
Type: Object
The stubbed object.

stubbedMethodName
Type: String
The name of the invoked method.

returnType
Type: System.Type
The return type of the invoked method.

listOfParamTypes
Type: ListYSTEM.Type>
A list of the parameter types of the invoked method.

listOfParamNames
Type: List<String>
A list of the parameter names of the invoked method.

listOfArgs
Type: List<Object>
The actual argument values passed into this method at runtime.

Return Value
Type: Object
Usage
You can use the parameters passed into this method to identify which method on the stubbed object was invoked. Then you can define the behavior for each identified method.

SEE ALSO:
Apex Developer Guide: Build a Mocking Framework with the Stub API

System Class
Contains methods for system operations, such as writing debug messages and scheduling jobs.

Namespace
System

System Methods
The following are methods for System. All methods are static.

IN THIS SECTION:
abortJob(jobId)
Stops the specified job. The stopped job is still visible in the job queue in the Salesforce user interface.
assert(condition, msg)
Asserts that the specified condition is true. If it isn’t, a fatal error is returned that causes code execution to halt.
assertEquals(expected, actual, msg)
Asserts that the first two arguments are the same. If they aren’t, a fatal error is returned that causes code execution to halt.
assertNotEquals(expected, actual, msg)
Asserts that the first two arguments are different. If they’re the same, a fatal error is returned that causes code execution to halt.
currentPageReference()
Returns a reference to the current page. This is used with Visualforce pages.
currentTimeMillis()
Returns the current time in milliseconds, which is expressed as the difference between the current time and midnight, January 1, 1970 UTC.
debug(msg)
Writes the specified message, in string format, to the execution debug log. The DEBUG log level is used.
debug(logLevel, msg)
Writes the specified message, in string format, to the execution debug log with the specified log level.
enqueueJob(queueableObj)
Adds a job to the Apex job queue that corresponds to the specified queueable class and returns the job ID.
enqueueJob(queueable, delay)
Adds a job to the Apex job queue that corresponds to the specified queueable class and returns the job ID. The job is scheduled with a specified minimum delay (0–10 minutes). The delay is ignored during Apex testing.
enqueueJob(queueable, asyncOptions)
Add a job to the Apex job queue that corresponds to the specified queueable class and returns the job ID. Specify a unique signature for your queueable job, the maximum stack depth or the minimum queue delay in the asyncOptions parameter.

equals(obj1, obj2)
Returns true if both arguments are equal. Otherwise, returns false.

getAddressReadWriteMode()
Returns the read write mode set for an organization during Salesforce.com upgrades and downtimes.

getQuiddityShortCode(QuiddityValue)
Returns the short code for the Quiddity value of the current Request object.

hashCode(obj)
Returns the hash code of the specified object.

isBatch()
Returns true if a batch Apex job invoked the executing code, or false if not. In API version 35.0 and earlier, also returns true if a queueable Apex job invoked the code.

isFunctionCallback()
Returns true if an asynchronous Salesforce Function callback invoked the executing code, or false if not. Available in API version 51.0 and later.

isFuture()
Returns true if the currently executing code is invoked by code contained in a method annotated with future; false otherwise.

isQueueable()
Returns true if a queueable Apex job invoked the executing code. Returns false if not, including if a batch Apex job or a future method invoked the code.

isRunningElasticCompute()
Reserved for future use.

isScheduled()
Returns true if the currently executing code is invoked by a scheduled Apex job; false otherwise.

movePassword(targetUserId,sourceUserId)
Moves the specified user’s password to a different user.

now()
Returns the current date and time in the GMT time zone.

process(workItemIds, action, comments, nextApprover)
Processes the list of work item IDs.

purgeOldAsyncJobs(dt)
Deletes asynchronous Apex job records for jobs that have finished execution before the specified date with a Completed, Aborted, or Failed status, and returns the number of records deleted.

requestVersion()
Returns a two-part version that contains the major and minor version numbers of a package.

resetPassword(userId, sendUserEmail)
Resets the password for the specified user.
resetPasswordWithEmailTemplate(userId, sendUserEmail, emailTemplateName)
Resets the user’s password and sends an email to the user with their new password. You specify the email template that is sent to the specified user. Use this method for external users of Experience Cloud sites.

runAs(version)
Changes the current package version to the package version specified in the argument.

runAs(userSObject)
Changes the current user to the specified user.

schedule(jobName, cronExpression, schedulableClass)
Use schedule with an Apex class that implements the Schedulable interface to schedule the class to run at the time specified by a Cron expression.

scheduleBatch(batchable, jobName, minutesFromNow)
Schedules a batch job to run once in the future after the specified time interval and with the specified job name.

scheduleBatch(batchable, jobName, minutesFromNow, scopeSize)
Schedules a batch job to run once in the future after the specified the time interval, with the specified job name and scope size. Returns the scheduled job ID (CronTrigger ID).

setPassword(userId, password)
Sets the password for the specified user.

submit(workItemIds, comments, nextApprover)
Submits the processed approvals. The current user is the submitter and the entry criteria is evaluated for all processes applicable to the current user.

today()
Returns the current date in the current user’s time zone.

abortJob(jobId)

Stop the specified job. The stopped job is still visible in the job queue in the Salesforce user interface.

**Signature**

```java
public static Void abortJob(String jobId)
```

**Parameters**

`jobId`
Type: String

The `jobId` is the ID associated with either AsyncApexJob or CronTrigger.

**Return Value**

Type: Void

**Usage**

The following methods return the job ID that can be passed to `abortJob`.

- **System.schedule method**—returns the CronTrigger object ID associated with the scheduled job as a string.
• **SchedulableContext.getTriggerId method**—returns the CronTrigger object ID associated with the scheduled job as a string.

• **getJobId method**—returns the AsyncApexJob object ID associated with the batch job as a string.

• **Using Batch Apex**
  - **Database.executeBatch method**—returns the AsyncApexJob object ID associated with the batch job as a string.

---

**assert(condition, msg)**

Asserts that the specified condition is true. If it isn’t, a fatal error is returned that causes code execution to halt.

**Important:** We recommend that you use the methods of the Assert Class rather than this method. The System.Assert class provides methods that handle all types of logical assertions and comparisons, which improve the clarity of your Apex code.

**Signature**

```
public static Void assert(Boolean condition, Object msg)
```

**Parameters**

- **condition**
  - Type: Boolean

- **msg**
  - Type: Object
  - (Optional) Custom message returned as part of the error message.

**Return Value**

Type: Void

**Usage**

You can’t catch an assertion failure using a try/catch block even though it’s logged as an exception.

---

**assertEquals(expected, actual, msg)**

Asserts that the first two arguments are the same. If they aren’t, a fatal error is returned that causes code execution to halt.

**Important:** We recommend that you use the methods of the Assert Class rather than this method. The System.Assert class provides methods that handle all types of logical assertions and comparisons, which improve the clarity of your Apex code.

**Signature**

```
public static Void assertEquals(Object expected, Object actual, Object msg)
```

**Parameters**

- **expected**
  - Type: Object
  - Specifies the expected value.
actual
  Type: Object
  Specifies the actual value.

msg
  Type: Object
  (Optional) Custom message returned as part of the error message.

Return Value
Type: Void

Usage
You can’t catch an assertion failure using a try/catch block even though it’s logged as an exception.

assertNotEquals(expected, actual, msg)
Asserts that the first two arguments are different. If they’re the same, a fatal error is returned that causes code execution to halt.

Important: We recommend that you use the methods of the Assert Class rather than this method. The System.Assert class provides methods that handle all types of logical assertions and comparisons, which improve the clarity of your Apex code.

Signature
public static Void assertNotEquals(Object expected, Object actual, Object msg)

Parameters
expected
  Type: Object
  Specifies the expected value.

actual
  Type: Object
  Specifies the actual value.

msg
  Type: Object
  (Optional) Custom message returned as part of the error message.

Return Value
Type: Void

Usage
You can’t catch an assertion failure using a try/catch block even though it’s logged as an exception.
currentPageReference()
Returns a reference to the current page. This is used with Visualforce pages.

Signature
public static System.PageReference currentPageReference()

Return Value
Type: System.PageReference

Usage
For more information, see PageReference Class.

currentTimeMillis()
Returns the current time in milliseconds, which is expressed as the difference between the current time and midnight, January 1, 1970 UTC.

Signature
public static Long currentTimeMillis()

Return Value
Type: Long

debug (msg)
Writes the specified message, in string format, to the execution debug log. The DEBUG log level is used.

Signature
public static Void debug(Object msg)

Parameters
msg
  Type: Object

Return Value
Type: Void

Usage
If the msg argument is not a string, the debug method calls String.valueOf to convert it into a string. The String.valueOf method calls the toString method on the argument, if available, or any overridden toString method if the argument is a user-defined type. Otherwise, if no toString method is available, it returns a string representation of the argument.
If the log level for Apex Code is set to `DEBUG` or higher, the message of this debug statement will be written to the debug log.

Note that when a map or set is printed, the output is sorted in key order and is surrounded with square brackets (```[]```). When an array or list is printed, the output is enclosed in parentheses (```()```).

Note: Calls to `System.debug` are not counted as part of Apex code coverage.

For more information on log levels, see Debug Log Levels in the Salesforce online help.

`debug(logLevel, msg)`

Writes the specified message, in string format, to the execution debug log with the specified log level.

**Signature**

```java
public static Void debug(LoggingLevel logLevel, Object msg)
```

**Parameters**

- `logLevel`
  - Type: `LoggingLevel Enum`
  - The logging level to set for this method.

- `msg`
  - Type: Object
  - The message or object to write in string format to the execution debug log.

**Return Value**

Type: Void

**Usage**

If the `msg` argument is not a string, the `debug` method calls `String.valueOf` to convert it into a string. The `String.valueOf` method calls the `toString` method on the argument, if available, or any overridden `toString` method if the argument is a user-defined type. Otherwise, if no `toString` method is available, it returns a string representation of the argument.

Note: Calls to `System.debug` are not counted as part of Apex code coverage.

For more information on log levels, see Debug Log Levels in the Salesforce online help.

`enqueueJob(queueableObj)`

Adds a job to the Apex job queue that corresponds to the specified queueable class and returns the job ID.

**Signature**

```java
public static ID enqueueJob(Object queueableObj)
```
**Parameters**

`queueableObj`

*Type: Object*

An instance of the class that implements the Queueable Interface.

**Return Value**

*Type: ID*

The job ID, which corresponds to the ID of an AsyncApexJob record.

**Usage**

To add a job for asynchronous execution, call `System.enqueueJob` by passing in an instance of your class implementation of the Queueable Interface for execution as follows:

```java
ID jobID = System.enqueueJob(new MyQueueableClass());
```

For more information about Queueable Apex, including information about limits, see Queueable Apex.

**enqueueJob(queueable, delay)**

Adds a job to the Apex job queue that corresponds to the specified queueable class and returns the job ID. The job is scheduled with a specified minimum delay (0–10 minutes). The delay is ignored during Apex testing.

**Signature**

```java
public static Id enqueueJob(Object queueable, Integer delay)
```

**Parameters**

`queueable`

*Type: Object*

An instance of the class that implements the Queueable Interface.

`delay`

*Type: Integer*

The minimum delay (0–10 minutes) before the queueable job is scheduled for execution.

The delay is ignored during Apex testing.

⚠️ **Warning:** When you set the delay to 0 (zero), the Queueable job is run as quickly as possible. With chained queueable jobs, implement a mechanism to slow down or halt the job if necessary. Without such a fail-safe mechanism in place, you can rapidly reach the daily async Apex limit.

**Return Value**

*Type: Id*

The job ID, which corresponds to the ID of an AsyncApexJob record.
Example

This example adds a job for delayed asynchronous execution by passing in an instance of your class implementation of the Queueable interface for execution. There's a minimum delay of 5 minutes before the job is executed.

```
Integer delayInMinutes = 5;
ID jobID = System.enqueueJob(new MyQueueableClass(), delayInMinutes);
```

For more information about Queueable Apex, including information about limits, see Queueable Apex.

enqueueJob (queueable, asyncOptions)

Adds a job to the Apex job queue that corresponds to the specified queueable class and returns the job ID. Specify a unique signature for your queueable job, the maximum stack depth or the minimum queue delay in the asyncOptions parameter.

Signature

```
public static Id enqueueJob(Object queueable, Object asyncOptions)
```

Parameters

- **queueable**
  - Type: Object
  - An instance of the class that implements the Queueable Interface.

- **asyncOptions**
  - Type: AsyncOptions
  - Specify a unique signature for your queueable job, the maximum stack depth, or a minimum queue delay in the AsyncOptions class properties.

Return Value

Type: Id

The job ID, which corresponds to the ID of an AsyncApexJob record.

Usage

The System.AsyncInfo class methods help you determine if maximum stack depth is set in your Queueable request and get the stack depths and queue delay for queueables that are currently running. Use information about the current queueable execution to make decisions on adjusting delays on subsequent calls.

These are methods in the System.AsyncInfo class.

- hasMaxStackDepth()
- getCurrentQueueableStackDepth()
- getMaximumQueueableStackDepth()
- getMinimumQueueableDelayInMinutes()

For more information about Queueable Apex, including information about limits, see Queueable Apex.
equals(obj1, obj2)

Returns true if both arguments are equal. Otherwise, returns false.

Signature

public static Boolean equals(Object obj1, Object obj2)

Parameters

obj1
Type: Object
Object being compared.

obj2
Type: Object
Object to compare with the first argument.

Return Value

Type: Boolean

Usage

obj1 and obj2 can be of any type. They can be values, or object references, such as sObjects and user-defined types.

The comparison rules for System.equals are identical to the ones for the == operator. For example, string comparison is case insensitive. For information about the comparison rules, see the == operator.

getApplicationReadWriteMode()

Returns the read write mode set for an organization during Salesforce.com upgrades and downtimes.

Signature

public static System.ApplicationReadWriteMode getApplicationReadWriteMode()

Return Value

Type: System.ApplicationReadWriteMode

Valid values are:

- DEFAULT
- READ_ONLY

Using the System.ApplicationReadWriteMode Enum

Use the System.ApplicationReadWriteMode enum returned by the getApplicationReadWriteMode to programmatically determine if the application is in read-only mode during Salesforce upgrades and downtimes.

Valid values for the enum are:

- DEFAULT
public class myClass {
    public static void execute() {
        ApplicationReadWriteMode mode = System.getApplicationReadWriteMode();

        if (mode == ApplicationReadWriteMode.READ_ONLY) {
            // Do nothing. If DML operation is attempted in readonly mode,
            // InvalidReadOnlyUserDmlException will be thrown.
        } else if (mode == ApplicationReadWriteMode.DEFAULT) {
            Account account = new Account(name = 'my account');
            insert account;
        }
    }
}

getQuiddityShortCode(QuiddityValue)
Returns the short code for the Quiddity value of the current Request object.

Signature
public String getQuiddityShortCode(System.Quiddity QuiddityValue)

Parameters
QuiddityValue
    Type: System.Quiddity
    The Quiddity enum value that has an associated short code. This short code is used in Event Monitoring logs. For more information, see Apex Execution Event Type.

Return Value
Type: String

hashCode(obj)
Returns the hash code of the specified object.

Signature
public static Integer hashCode(Object obj)

Parameters
obj
    Type: Object
    The object to get the hash code for. This parameter can be of any type, including values or object references, such as sObjects or user-defined types.
Return Value
Type: Integer

Versioned Behavior Changes
In API version 51.0 and later, the `hashCode()` method returns the same hashCode for identical Id values. In API version 50.0 and earlier, identical Id values didn't always generate the same hashCode value.

`isBatch()`
Returns `true` if a batch Apex job invoked the executing code, or `false` if not. In API version 35.0 and earlier, also returns `true` if a queueable Apex job invoked the code.

Signature
`public static Boolean isBatch()`

Return Value
Type: Boolean

Usage
A batch Apex job can't invoke a future method. Before invoking a future method, use `isBatch()` to check whether the executing code is a batch Apex job.

`isFunctionCallback()`
Returns `true` if an asynchronous Salesforce Function callback invoked the executing code, or `false` if not. Available in API version 51.0 and later.

Signature
`public static Boolean isFunctionCallback()`

Return Value
Type: Boolean

Usage
Use this method to determine if the Apex code is being invoked as part of a callback from an asynchronous Salesforce Functions invocation. For more details on invoking Salesforce Functions from Apex, see Functions Namespace.

`isFuture()`
Returns `true` if the currently executing code is invoked by code contained in a method annotated with `future`; `false` otherwise.
**Signature**

```
public static Boolean isFuture()
```

**Return Value**

Type: Boolean

**Usage**

Since a future method can’t be invoked from another future method, use this method to check if the current code is executing within the context of a future method before you invoke a future method.

**isQueueable()**

Returns true if a queueable Apex job invoked the executing code. Returns false if not, including if a batch Apex job or a future method invoked the code.

**Signature**

```
public static Boolean isQueueable()
```

**Return Value**

Type: Boolean

**Usage**

```java
public class SimpleQueueable implements Queueable {

    String name;

    public SimpleQueueable(String name) {
        this.name = name;
        System.assert(!System.isQueueable()); //Should return false
    }

    public void execute(QueueableContext ctx) {
        Account testAccount = new Account();
        testAccount.name = 'testAcc';
        insert(testAccount);
        System.assert(System.isQueueable()); //Should return true
    }
}
```

```java
global class ComplexBatch implements Database.Batchable<SObject> {

    global Database.QueryLocator start(Database.BatchableContext info) {
        System.assert(!System.isQueueable()); //Should return false
        return Database.getQueryLocator(([SELECT Id, Name FROM Account LIMIT 1]);
    }

    global void execute(Database.BatchableContext info, SObject[] scope) {
```
isRunningElasticCompute()
Reserved for future use.

Signature
public static Boolean isRunningElasticCompute()

Return Value
Type: Boolean

isScheduled()
Returns true if the currently executing code is invoked by a scheduled Apex job; false otherwise.

Signature
public static Boolean isScheduled()

Return Value
Type: Boolean

movePassword(targetUserId, sourceUserId)
Moves the specified user's password to a different user.

Signature
public static Void movePassword(ID targetUserId, ID sourceUserId)

Parameters

targetUserId
Type: ID
The user that the password is moved to.

sourceUserId
Type: ID
The user that the password is moved from.
Return Value
Type: Void

Usage
Moving a password simplifies converting a user to another type of user, such as when converting an external user to a user with less restrictive access. If you require access to the movePassword method, contact Salesforce.

Keep in mind these requirements.
- The targetUserId, sourceUserId, and user performing the move operation must all belong to the same Salesforce org.
- The targetUserId and the sourceUserId cannot be the same as the user performing the move operation.
- A user without a password can't be specified as the sourceUserId. For example, a source user who has already had their password moved is left without a password. That user can’t be a source user again.

After the password is moved:
- The target user can log in with the password.
- The source user no longer has a password. To enable logins for this user, a password reset is required.

now()
Returns the current date and time in the GMT time zone.

Signature
public static Datetime now()

Return Value
Type: Datetime

process(workItemIds, action, comments, nextApprover)
Processes the list of work item IDs.

Signature
public static List<Id> process(List<Id> workItemIds, String action, String comments, String nextApprover)

Parameters
workItemIds
  Type: List<Id>
action
  Type: String
comments
  Type: String
nextApprover
  Type: String
Return Value
Type: List<Id>

**purgeOldAsyncJobs (dt)**
Deletes asynchronous Apex job records for jobs that have finished execution before the specified date with a Completed, Aborted, or Failed status, and returns the number of records deleted.

**Signature**

```java
public static Integer purgeOldAsyncJobs(Date dt)
```

**Parameters**

dt
  Type: Date
  Specifies the date up to which old records are deleted. The date comparison is based on the `CompletedDate` field of `AsyncApexJob`, which is in the GMT time zone.

**Return Value**
Type: Integer

**Usage**
Asynchronous Apex job records are records in `AsyncApexJob`.
The system cleans up asynchronous job records for jobs that have finished execution and are older than seven days. You can use this method to further reduce the size of `AsyncApexJob` by cleaning up more records.
Each execution of this method counts as a single row against the governor limit for DML statements.

**Example**
This example shows how to delete all job records for jobs that have finished before today’s date.

```java
Integer count = System.purgeOldAsyncJobs(Date.today());
System.debug('Deleted ' + count + ' old jobs.);
```

**requestVersion()**
Returns a two-part version that contains the major and minor version numbers of a package.

**Signature**

```java
public static System.Version requestVersion()
```

**Return Value**
Type: System.Version
Usage
Using this method, you can determine the version of an installed instance of your package from which the calling code is referencing your package. Based on the version that the calling code has, you can customize the behavior of your package code.

The requestVersion method isn’t supported for unmanaged packages. If you call it from an unmanaged package, an exception will be thrown.

```
resetPassword(userId, sendUserEmail)
```

Resets the password for the specified user.

**Signature**

```
public static System.ResetPasswordResult resetPassword(ID userId, Boolean sendUserEmail)
```

**Parameters**

- `userId` 
  Type: ID

- `sendUserEmail` 
  Type: Boolean

**Return Value**

Type: System.ResetPasswordResult

**Usage**

When the user logs in with the new password, they are prompted to enter a new password, and to select a security question and answer if they haven’t already. If you specify `true` for `sendUserEmail`, the user is sent an email notifying them that their password was reset. A link to sign onto Salesforce using the new password is included in the email. Use `setPassword(userId, password)` if you don’t want the user to be prompted to enter a new password when they log in.

⚠️ **Warning**: Be careful with this method, and do not expose this functionality to end-users.

```
resetPasswordWithEmailTemplate(userId, sendUserEmail, emailTemplateName)
```

Resets the user’s password and sends an email to the user with their new password. You specify the email template that is sent to the specified user. Use this method for external users of Experience Cloud sites.

**Signature**

```
public static System.ResetPasswordResult resetPasswordWithEmailTemplate(Id userId, Boolean sendUserEmail, String emailTemplateName)
```

**Parameters**

- `userId` 
  Type: Id

  The ID of the user whose password was reset.
sendUserEmail
    Type: Boolean

templateName
    Type: String

Return Value
    Type: System.ResetPasswordResult

Usage
If you specify true for sendUserEmail, specify the email template that is sent to the user notifying them that their password was
reset. When the user logs in with the new password in the email, they are prompted to enter a new password. A link to sign onto
Salesforce using the new password is included in the email. Use setPassword(userId, password) if you don’t want the user
to be prompted to enter a new password when they log in.

⚠️ Warning: Be careful with this method, and do not expose this functionality to end-users.

runAs(version)
Changes the current package version to the package version specified in the argument.

Signature
    public static Void runAs(System.Version version)

Parameters
    version
        Type: System.Version

Return Value
    Type: Void

Usage
A package developer can use Version methods to continue to support existing behavior in classes and triggers in previous package
versions while continuing to evolve the code. Apex classes and triggers are saved with the version settings for each installed managed
package that the class or trigger references.

This method is used for testing your component behavior in different package versions that you upload to the AppExchange. This method
effectively sets a two-part version consisting of major and minor numbers in a test method so that you can test the behavior for different
package versions.

You can only use runAs in a test method. There is no limitation to the number of calls to this method in a transaction. For sample
usage of this method, see Testing Behavior in Package Versions.
**runAs (userSObject)**
Changes the current user to the specified user.

**Signature**
```
public static Void runAs(User userSObject)
```

**Parameters**
- **userSObject**
  Type: User

**Return Value**
Type: Void

**Usage**
All of the specified user’s record sharing is enforced during the execution of `runAs`. You can only use `runAs` in a test method. For more information, see Using the `runAs()` Method.

**Note:** The `runAs` method ignores user license limits. You can create new users with `runAs` even if your organization has no additional user licenses.

The `runAs` method implicitly inserts the user that is passed in as parameter if the user has been instantiated, but not inserted yet. You can also use `runAs` to perform mixed DML operations in your test by enclosing the DML operations within the `runAs` block. In this way, you bypass the mixed DML error that is otherwise returned when inserting or updating setup objects together with other sObjects. See sObjects That Cannot Be Used Together in DML Operations.

**Note:** Every call to `runAs` counts against the total number of DML statements issued in the process.

**schedule(jobName, cronExpression, schedulableClass)**
Use `schedule` with an Apex class that implements the `Schedulable` interface to schedule the class to run at the time specified by a Cron expression.

**Signature**
```
public static String schedule(String jobName, String cronExpression, Object schedulableClass)
```

**Parameters**
- **jobName**
  Type: String
- **cronExpression**
  Type: String
- **schedulableClass**
  Type: Object
Return Value
Type: String
Returns the scheduled job ID (CronTrigger ID).

Usage
Use extreme care if you’re planning to schedule a class from a trigger. You must be able to guarantee that the trigger won’t add more scheduled classes than the limit. In particular, consider API bulk updates, import wizards, mass record changes through the user interface, and all cases where more than one record can be updated at a time. Use the abortJob method to stop the job after it has been scheduled.

Note: Salesforce schedules the class for execution at the specified time. Actual execution may be delayed based on service availability.

Using the System.Schedule Method
After you implement a class with the Schedulable interface, use the System.Schedule method to execute it. The scheduler runs as system—all classes are executed, whether or not the user has permission to execute the class.

Note: Use extreme care if you’re planning to schedule a class from a trigger. You must be able to guarantee that the trigger won’t add more scheduled classes than the limit. In particular, consider API bulk updates, import wizards, mass record changes through the user interface, and all cases where more than one record can be updated at a time.

The System.Schedule method takes three arguments: a name for the job, an expression used to represent the time and date the job is scheduled to run, and the name of the class. This expression has the following syntax:

Seconds Minutes Hours Day_of_month Month Day_of_week Optional_year

Note: Salesforce schedules the class for execution at the specified time. Actual execution may be delayed based on service availability.

The System.Schedule method uses the user’s timezone for the basis of all schedules.

The following are the values for the expression:

<table>
<thead>
<tr>
<th>Name</th>
<th>Values</th>
<th>Special Characters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seconds</td>
<td>0–59</td>
<td>None</td>
</tr>
<tr>
<td>Minutes</td>
<td>0–59</td>
<td>None</td>
</tr>
<tr>
<td>Hours</td>
<td>0–23</td>
<td>, - * /</td>
</tr>
<tr>
<td>Month</td>
<td>1–12 or the following:</td>
<td>, - * /</td>
</tr>
<tr>
<td></td>
<td>• JAN</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• FEB</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• MAR</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• APR</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• MAY</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• JUN</td>
<td></td>
</tr>
</tbody>
</table>
## Special Characters

### Values

- JAN
- MAR
- APR
- MAY
- JUN
- JUL
- AUG
- SEP
- OCT
- NOV
- DEC

### Special Characters

- \,
- -
- *
- ?
- / (starting on the first of the month)
- L
- #

### Description

1. **\,** Delimits values. For example, use JAN, MAR, APR to specify more than one month.
2. **-** Specifies a range. For example, use JAN-MAR to specify more than one month.
3. ***\*** Specifies all values. For example, if Month is specified as *, the job is scheduled for every month.
4. **?** Specifies no specific value. This is only available for Day_of_month and Day_of_week, and is generally used when specifying a value for one and not the other.
5. **/** Specifies increments. The number before the slash specifies when the intervals will begin, and the number after the slash is the interval amount. For example, if you specify 1/5 for Day_of_month, the Apex class runs every fifth day of the month, starting on the first of the month.
6. **L** Specifies the end of a range (last). This is only available for Day_of_month and Day_of_week. When used with Day_of_month, L always means the last day of the month, such as January 31, February 29 for leap years, and so on. When used with Day_of_week by itself, it always means 7 or SAT. When used with a Day_of_week value, it means the last of that type of day in the month. For example, if you specify 2L, you are specifying the last Monday of the month. Do not use a range of values with L as the results might be unexpected.
<table>
<thead>
<tr>
<th>Special Character</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>W</td>
<td>Specifies the nearest weekday (Monday-Friday) of the given day. This is only available for <code>Day_of_month</code>. For example, if you specify 20W, and the 20th is a Saturday, the class runs on the 19th. If you specify 1W, and the first is a Saturday, the class does not run in the previous month, but on the third, which is the following Monday.</td>
</tr>
</tbody>
</table>

**Tip:** Use the L and W together to specify the last weekday of the month.

| #                 | Specifies the `n`th day of the month, in the format `weekday#day_of_month`. This is only available for `Day_of_week`. The number before the # specifies weekday (SUN-SAT). The number after the # specifies the day of the month. For example, specifying 2#1 means the class runs on the first Monday of every month. |

The following are some examples of how to use the expression.

<table>
<thead>
<tr>
<th>Expression</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 13 * * ?</td>
<td>Class runs every day at 1 PM.</td>
</tr>
<tr>
<td>0 0 22 ? * 6L</td>
<td>Class runs the last Friday of every month at 10 PM.</td>
</tr>
<tr>
<td>0 0 10 ? * MON-FRI</td>
<td>Class runs Monday through Friday at 10 AM.</td>
</tr>
<tr>
<td>0 0 20 * * ? 2010</td>
<td>Class runs every day at 8 PM during the year 2010.</td>
</tr>
</tbody>
</table>

In the following example, the class `proschedule` implements the `Schedulable` interface. The class is scheduled to run at 8 AM, on the 13 February.

```java
proschedule p = new proschedule();
   String sch = '0 0 8 13 2 ?';
   system.schedule('One Time Pro', sch, p);
```

**scheduleBatch(batchable, jobName, minutesFromNow)**

Schedules a batch job to run once in the future after the specified time interval and with the specified job name.

**Signature**

```java
public static String scheduleBatch(Database.Batchable batchable, String jobName, Integer minutesFromNow)
```

**Parameters**

- **batchable**
  
  Type: `Database.Batchable`
  
  An instance of a class that implements the `Database.Batchable` interface.

- **jobName**
  
  Type: `String`
The name if the job that this method will start.

`minutesFromNow`
Type: `Integer`

The time interval in minutes after which the job should start executing. This argument must be greater than zero.

**Return Value**
Type: `String`
The scheduled job ID (CronTrigger ID).

**Usage**

**Note:** Some things to note about `System.scheduleBatch`:

- When you call `System.scheduleBatch`, Salesforce schedules the job for execution at the specified time. Actual execution occurs at or after that time, depending on service availability.
- The scheduler runs as system—all classes are executed, whether the user has permission to execute the class, or not.
- When the job’s schedule is triggered, the system queues the batch job for processing. If Apex flex queue is enabled in your org, the batch job is added at the end of the flex queue. For more information, see Holding Batch Jobs in the Apex Flex Queue.
- All scheduled Apex limits apply for batch jobs scheduled using `System.scheduleBatch`. After the batch job is queued (with a status of Holding or Queued), all batch job limits apply and the job no longer counts toward scheduled Apex limits.
- After calling this method and before the batch job starts, you can use the returned scheduled job ID to abort the scheduled job using the `System.abortJob` method.

For an example, see Using Batch Apex.

```plaintext
scheduleBatch(batchable, jobName, minutesFromNow, scopeSize)
```

Schedules a batch job to run once in the future after the specified the time interval, with the specified job name and scope size. Returns the scheduled job ID (CronTrigger ID).

**Signature**

```plaintext
public static String scheduleBatch(Database.Batchable batchable, String jobName, Integer minutesFromNow, Integer scopeSize)
```

**Parameters**

- `batchable`
  Type: `Database.Batchable`
  The batch class that implements the `Database.Batchable` interface.

- `jobName`
  Type: `String`
  The name of the job that this method will start.

- `minutesFromNow`
  Type: `Integer`
The time interval in minutes after which the job should start executing.

`scopeSize`
Type: `Integer`

The number of records that should be passed to the batch `execute` method.

**Return Value**
Type: `String`

**Usage**

*Note:* Some things to note about `System.scheduleBatch`:
- When you call `System.scheduleBatch`, Salesforce schedules the job for execution at the specified time. Actual execution occurs at or after that time, depending on service availability.
- The scheduler runs as system— all classes are executed, whether the user has permission to execute the class, or not.
- When the job’s schedule is triggered, the system queues the batch job for processing. If Apex flex queue is enabled in your org, the batch job is added at the end of the flex queue. For more information, see Holding Batch Jobs in the Apex Flex Queue.
- All scheduled Apex limits apply for batch jobs scheduled using `System.scheduleBatch`. After the batch job is queued (with a status of `Holding` or `Queued`), all batch job limits apply and the job no longer counts toward scheduled Apex limits.
- After calling this method and before the batch job starts, you can use the returned scheduled job ID to abort the scheduled job using the `System.abortJob` method.

For an example, see Using the `System.scheduleBatch` Method.

**setPassword(userId, password)**

Sets the password for the specified user.

**Signature**

```java
public static Void setPassword(ID userId, String password)
```

**Parameters**

`userId`
Type: `ID`

`password`
Type: `String`

**Return Value**
Type: `Void`

**Usage**

- If a security question hasn’t been previously configured, a user who logs in with a new password that was set using `setPassword` is redirected to the “Change Your Password” page.
• Use `resetPassword(userId, sendUserEmail)` if you want the user to go through the reset process and create their own password.

⚠️ **Warning:** Be careful with this method, and don’t expose this functionality to end users.

### submit(workItemIds, comments, nextApprover)

Submits the processed approvals. The current user is the submitter and the entry criteria is evaluated for all processes applicable to the current user.

**Signature**

```java
public static List<ID> submit(List<ID> workItemIds, String comments, String nextApprover)
```

**Parameters**

- `workItemIds`  
  Type: `List<ID>`
- `comments`  
  Type: `String`
- `nextApprover`  
  Type: `String`

**Return Value**

Type: `List<ID>`

**Usage**

For enhanced submit and evaluation features, see the `ProcessSubmitRequest` class.

### today()

Returns the current date in the current user’s time zone.

**Signature**

```java
public static Date today()
```

**Return Value**

Type: `Date`

---

**Test Class**

Contains methods related to Apex tests.
Namespace

System

Test Methods

The following are methods for Test. All methods are static.

IN THIS SECTION:

- **calculatePermissionSetGroup(psgIds)**
  Calculates aggregate permissions in specified permission set groups for testing.

- **calculatePermissionSetGroup(psgId)**
  Calculates aggregate permissions in a specified permission set group for testing.

- **clearApexPageMessages()**
  Clear the messages on a Visualforce page while executing Apex test methods.

- **createStub(parentType, stubProvider)**
  Creates a stubbed version of an Apex class that you can use for testing. This method is part of the Apex stub API. You can use it with the System.StubProvider interface to create a mocking framework.

- **enableChangeDataCapture()**
  Use this method in an Apex test so that change event notifications are generated for all supported Change Data Capture entities. Call this method at the beginning of your test before performing DML operations and calling Test.getEventBus().deliver();.

- **enqueueBatchJobs(numberOfJobs)**
  Adds the specified number of jobs with no-operation contents to the test-context queue. It first fills the test batch queue, up to the maximum 5 jobs, and then places jobs in the test flex queue. It throws a limit exception when the number of jobs in the test flex queue exceeds the allowed limit of 100 jobs.

- **getEventBus()**
  Returns an instance of the test event bus broker, which lets you operate on platform event or change event messages in an Apex test. For example, you can call Test.getEventBus().deliver() to deliver event messages.

- **getFlexQueueOrder()**
  Returns an ordered list of job IDs for jobs in the test-context flex queue. The job at index 0 is the next job slated to run. This method returns only test-context results, even if it’s annotated with @IsTest(SeeAllData=true).

- **getStandardPricebookId()**
  Returns the ID of the standard price book in the organization.

- **invokeContinuationMethod(controller, request)**
  Invokes the callback method for the specified controller and continuation in a test method.

- **isRunningTest()**
  Returns true if the currently executing code was called by code contained in a test method, false otherwise. Use this method if you need to run different code depending on whether it was being called from a test.

- **loadData(sObjectToken, resourceName)**
  Inserts test records from the specified static resource .csv file and for the specified sObject type, and returns a list of the inserted sObjects.
newSendEmailQuickActionDefaults(contextId, replyToId)
Creates a new QuickAction.SendEmailQuickActionDefaults instance for testing a class implementing the QuickActionQuickActionDefaultsHandler interface.

setContinuationResponse(requestLabel, mockResponse)
Sets a mock response for a continuation HTTP request in a test method.

setCreatedDate(recordId, createdDatetime)
Sets CreatedDate for a test-context sObject.

setCurrentPage(page)
A Visualforce test method that sets the current PageReference for the controller.

setCurrentPageReference(page)
A Visualforce test method that sets the current PageReference for the controller.

setFixedSearchResults(fixedSearchResults)
Defines a list of fixed search results to be returned by all subsequent SOSL statements in a test method.

setMock(interfaceType, instance)
Sets the response mock mode and instructs the Apex runtime to send a mock response whenever a callout is made through the HTTP classes or the auto-generated code from WSDLs.

setReadOnlyApplicationMode(applicationMode)
Sets the application mode for an organization to read-only in an Apex test to simulate read-only mode during Salesforce upgrades and downtimes. The application mode is reset to the default mode at the end of each Apex test run.

startTest()
Marks the point in your test code when your test actually begins. Use this method when you are testing governor limits.

stopTest()
Marks the point in your test code when your test ends. Use this method in conjunction with the startTest method.

testInstall(installImplementation, version, isPush)
Tests the implementation of the InstallHandler interface, which is used for specifying a post install script in packages. Tests run as the test initiator in the development environment.

testSandboxPostCopyScript(script, organizationId, sandboxId, sandboxName)
Tests the implementation of the SandboxPostCopy Interface, which is used for specifying a script to run at the completion of a Sandbox copy. Tests run as the test initiator in the development environment.

testSandboxPostCopyScript(script, organizationId, sandboxId, sandboxName, RunAsAutoProcUser)
Tests the implementation of the SandboxPostCopy Interface, which is used for specifying a script to run at the completion of a Sandbox copy. When RunAsAutoProcUser is true, tests run as Automated Process user in the development environment.

testUninstall(uninstallImplementation)
Tests the implementation of the UninstallHandler interface, which is used for specifying an uninstall script in packages. Tests run as the test initiator in the development environment.

calculatePermissionSetGroup(psgIds)
Calculates aggregate permissions in specified permission set groups for testing.

**Signature**

```java
public static void calculatePermissionSetGroup(List<String> psgIds)
```
Parameters

*psgIds*
  Type: `List<String>`
  A list of IDs for permission set groups.

Return Value

Type: `void`

`calculatePermissionSetGroup(psgId)`
Calculates aggregate permissions in a specified permission set group for testing.

Signature

```java
public static void calculatePermissionSetGroup(String psgId)
```

Parameters

*psgId*
  Type: `String`
  A single ID for a specified permission set group.

Return Value

Type: `void`

`clearApexPageMessages()`
Clear the messages on a Visualforce page while executing Apex test methods.

Signature

```java
public static void clearApexPageMessages()
```

Return Value

Type: `void`

Usage

This method may only be used in tests.

Example:

```java
@isTest
static void clearMessagesTest() {
    Test.setCurrentPage(new PageReference('/'));
    ApexPages.addMessage(
        new ApexPages.Message(ApexPages.Severity.WARNING, 'Sample Warning'))
```
createStub(parentType, stubProvider)

Creates a stubbed version of an Apex class that you can use for testing. This method is part of the Apex stub API. You can use it with the System.StubProvider interface to create a mocking framework.

Signature

public static Object createStub(System.Type parentType, System.StubProvider stubProvider)

Parameters

parentType
Type: System.Type
The type of the Apex class to be stubbed.

stubProvider
System.StubProvider
An implementation of the StubProvider interface.

Return Value

Type: Object
Returns the stubbed object to use in testing.

Usage

The createStub() method works together with the System.StubProvider interface. You define the behavior of the stubbed object by implementing the StubProvider interface. Then you create a stubbed object using the createStub() method. When you invoke methods on the stubbed object, the handleMethodCall() method of the StubProvider interface is called to perform the behavior of the stubbed method.

SEE ALSO:
Apex Developer Guide: Build a Mocking Framework with the Stub API

enableChangeDataCapture()

Use this method in an Apex test so that change event notifications are generated for all supported Change Data Capture entities. Call this method at the beginning of your test before performing DML operations and calling Test.getEventBus().deliver();.

Signature

public static void enableChangeDataCapture()
**Return Value**

Type: void

**Usage**

The `enableChangeDataCapture()` method ensures that Apex tests can fire change event triggers regardless of the entities selected in Setup in the Change Data Capture page. The `enableChangeDataCapture()` method doesn't affect the entities selected in Setup.

SEE ALSO:

*Change Data Capture Developer Guide*

---

**enqueueBatchJobs(numberOfJobs)**

Adds the specified number of jobs with no-operation contents to the test-context queue. It first fills the test batch queue, up to the maximum 5 jobs, and then places jobs in the test flex queue. It throws a limit exception when the number of jobs in the test flex queue exceeds the allowed limit of 100 jobs.

**Signature**

```java
public static List<Id> enqueueBatchJobs(Integer numberOfJobs)
```

**Parameters**

`numberOfJobs`

Type: `Integer`

Number of test jobs to enqueue.

**Return Value**

Type: `List<Id>`

A list of IDs of enqueued test jobs.

**Usage**

Use this method to reduce testing time. Instead of using your org's real batch jobs for testing, you can use this method to simulate batch-job enqueueing. Using `enqueueBatchJobs(numberOfJobs)` is faster than enqueuing real batch jobs.

---

**getEventBus()**

Returns an instance of the test event bus broker, which lets you operate on platform event or change event messages in an Apex test. For example, you can call `Test.getEventBus().deliver()` to deliver event messages.

**Signature**

```java
public static EventBus.TestBroker getEventBus()
```
Return Value
Type: `EventBus.TestBroker`
A broker for the test event bus.

Usage
Enclose `Test.getEventBus().deliver()` within the `Test.startTest()` and `Test.stopTest()` statement block.

```java
Test.startTest();
// Create test events
// ...
// Publish test events with EventBus.publish()
// ...
// Deliver test events
Test.getEventBus().deliver();
// Perform validation
// ...
Test.stopTest();
```

SEE ALSO:
Platform Events Developer Guide

---

### getFlexQueueOrder()

Returns an ordered list of job IDs for jobs in the test-context flex queue. The job at index 0 is the next job slated to run. This method returns only test-context results, even if it’s annotated with `@IsTest(SeeAllData=true)`.

**Signature**

```java
public static List<Id> getFlexQueueOrder()
```

**Return Value**

Type: `List<Id>`
An ordered list of IDs of the jobs in the test’s flex queue.

---

### getStandardPricebookId()

Returns the ID of the standard price book in the organization.

**Signature**

```java
public static Id getStandardPricebookId()
```

**Return Value**

Type: `Id`
The ID of the standard price book.
Usage

This method returns the ID of the standard price book in your organization regardless of whether the test can query organization data. By default, tests can’t query organization data unless they’re annotated with \@isTest(SeeAllData=true).

Creating price book entries with a standard price requires the ID of the standard price book. Use this method to get the standard price book ID so that you can create price book entries in your tests.

Example

This example creates some test data for price book entries. The test method in this example gets the standard price book ID and uses this ID to create a price book entry for a product with a standard price. Next, the test creates a custom price book and uses the ID of this custom price book to add a price book entry with a custom price.

```java
@isTest
class PriceBookTest {
    // Utility method that can be called by Apex tests to create price book entries.
    static testmethod void addPricebookEntries() {
        // First, set up test price book entries.
        // Insert a test product.
        Product2 prod = new Product2(Name = 'Laptop X200',
            Family = 'Hardware');
        insert prod;

        // Get standard price book ID.
        // This is available irrespective of the state of SeeAllData.
        Id pricebookId = Test.getStandardPricebookId();

        // 1. Insert a price book entry for the standard price book.
        PricebookEntry standardPrice = new PricebookEntry(
            Pricebook2Id = pricebookId, Product2Id = prod.Id,
            UnitPrice = 10000, IsActive = true);
        insert standardPrice;

        // Create a custom price book
        Pricebook2 customPB = new Pricebook2(Name='Custom Pricebook', isActive=true);
        insert customPB;

        // 2. Insert a price book entry with a custom price.
        PricebookEntry customPrice = new PricebookEntry(
            Pricebook2Id = customPB.Id, Product2Id = prod.Id,
            UnitPrice = 12000, IsActive = true);
        insert customPrice;

        // Next, perform some tests with your test price book entries.
    }
}
```

`invokeContinuationMethod(controller, request)`

Invokes the callback method for the specified controller and continuation in a test method.
Signature

public static Object invokeContinuationMethod(Object controller, Continuation request)

Parameters

controller
  Type: Object
  An instance of the controller class that invokes the continuation request.

request
  Type: Continuation
  The continuation that is returned by an action method in the controller class.

Return Value

Type: Object
The response of the continuation callback method.

Usage

Use the Test.setContinuationResponse and Test.invokeContinuationMethod methods to test continuations. In test context, callouts of continuations aren’t sent to the external service. By using these methods, you can set a mock response and cause the runtime to call the continuation callback method to process the mock response.

Call Test.setContinuationResponse before you call Test.invokeContinuationMethod. When you call Test.invokeContinuationMethod, the runtime executes the callback method that is associated with the continuation. The callback method processes the mock response that is set by Test.setContinuationResponse.

isRunningTest()

Returns true if the currently executing code was called by code contained in a test method, false otherwise. Use this method if you need to run different code depending on whether it was being called from a test.

Signature

public static Boolean isRunningTest()

Return Value

Type: Boolean

loadData(sObjectToken, resourceName)

Inserts test records from the specified static resource .csv file and for the specified sObject type, and returns a list of the inserted sObjects.

Signature

public static List<sObject> loadData(Schema.SObjectType sObjectToken, String resourceName)
Parameters

sObjectToken
Type: Schema.SObjectType
The sObject type for which to insert test records.

resourceName
Type: String
The static resource that corresponds to the .csv file containing the test records to load. The name is case insensitive.

Return Value
Type: List<sObject>

Usage
You must create the static resource prior to calling this method. The static resource is a comma-delimited file ending with a .csv extension. The file contains field names and values for the test records. The first line of the file must contain the field names and subsequent lines are the field values. To learn more about static resources, see "Defining Static Resources" in the Salesforce online help.

Once you create a static resource for your .csv file, the static resource will be assigned a MIME type. Supported MIME types are:

- text/csv
- application/vnd.ms-excel
- application/octet-stream
- text/plain

newSendEmailQuickActionDefaults(contextId, replyToId)
Creates a new QuickAction.SendEmailQuickActionDefaults instance for testing a class implementing the QuickAction.QuickActionDefaultsHandler interface.

Signature
public static QuickAction.SendEmailQuickActionDefaults newSendEmailQuickActionDefaults(ID contextId, ID replyToId)

Parameters

contextId
Type: Id
Parent record of the email message.

replyToId
Type: Id
Previous email message ID if this email message is a reply.

Return Value
Type: SendEmailQuickActionDefaults Class
The default values used for an email message quick action.
setContinuationResponse(requestLabel, mockResponse)
Sets a mock response for a continuation HTTP request in a test method.

Signature

```java
public static void setContinuationResponse(String requestLabel, System.HttpResponse mockResponse)
```

Parameters

- **requestLabel**
  - Type: `String`
  - The unique label that corresponds to the continuation HTTP request. This label is returned by `Continuation.addHttpRequest`.

- **mockResponse**
  - Type: `HttpResponse`
  - The fake response to be returned by `Test.invokeContinuationMethod`.

Return Value

Type: `void`

Usage

Use the `Test.setContinuationResponse` and `Test.invokeContinuationMethod` methods to test continuations. In test context, callouts of continuations aren’t sent to the external service. By using these methods, you can set a mock response and cause the runtime to call the continuation callback method to process the mock response.

Call `Test.setContinuationResponse` before you call `Test.invokeContinuationMethod`. When you call `Test.invokeContinuationMethod`, the runtime executes the callback method that is associated with the continuation. The callback method processes the mock response that is set by `Test.setContinuationResponse`.

setCreatedDate(recordId, createdDatetime)
Sets CreatedDate for a test-context sObject.

Signature

```java
public static void setCreatedDate(Id recordId, Datetime createdDatetime)
```

Parameters

- **recordId**
  - Type: `Id`
  - The ID of an sObject.

- **createdDatetime**
  - Type: `Datetime`
  - The value to assign to the sObject’s CreatedDate field.
Return Value
Type: void

Usage
All database changes are rolled back at the end of a test. You can’t use this method on records that existed before your test executed. You also can’t use setCreatedDate in methods annotated with @isTest(SeeAllData=true), because those methods have access to all data in your org. If you set CreatedDate to a future value, it can cause unexpected results. This method takes two parameters—an sObject ID and a Datetime value—neither of which can be null.

Insert your test record before you set its CreatedDate, as shown in this example.

```apex
@isTest
private class SetCreatedDateTest {
    static testMethod void testSetCreatedDate() {
        Account a = new Account(name='myAccount');
        insert a;
        Test.setCreatedDate(a.Id, DateTime.newInstance(2012,12,12));
        Test.startTest();
        Account myAccount = [SELECT Id, Name, CreatedDate FROM Account WHERE Name='myAccount' limit 1];
        System.assertEquals(myAccount.CreatedDate, DateTime.newInstance(2012,12,12));
        Test.stopTest();
    }
}
```

**setCurrentPage(page)**
A Visualforce test method that sets the current PageReference for the controller.

**Signature**
```apex```
public static Void setCurrentPage(PageReference page)
```apex```

**Parameters**
```apex```
page
Type: System.PageReference
```apex```

**Return Value**
Type: Void

**setCurrentPageReference(page)**
A Visualforce test method that sets the current PageReference for the controller.

**Signature**
```apex```
public static Void setCurrentPageReference(PageReference page)
```apex```
Parameters

page
Type: System.PageReference

Return Value
Type: Void

setFixedSearchResults(fixedSearchResults)
Defines a list of fixed search results to be returned by all subsequent SOSL statements in a test method.

Signature
public static Void setFixedSearchResults(ID[] fixedSearchResults)

Parameters

fixedSearchResults
Type: ID[]
The list of record IDs specified by opt_set_search_results replaces the results that would normally be returned by the SOSL queries if they were not subject to any WHERE or LIMIT clauses. If these clauses exist in the SOSL queries, they are applied to the list of fixed search results.

Return Value
Type: Void

Usage
If opt_set_search_results is not specified, all subsequent SOSL queries return no results.
For more information, see Dynamic SOSL.

setMock(interfaceType, instance)
Sets the response mock mode and instructs the Apex runtime to send a mock response whenever a callout is made through the HTTP classes or the auto-generated code from WSDLs.

Signature
public static Void setMock(Type interfaceType, Object instance)

Parameters

interfaceType
Type: System.Type

instance
Type: Object
Return Value
Type: Void

Usage

Note: To mock a callout if the code that performs the callout is in a managed package, call Test.setMock from a test method in the same package with the same namespace.

**setReadOnlyApplicationMode** *(applicationMode)*

Sets the application mode for an organization to read-only in an Apex test to simulate read-only mode during Salesforce upgrades and downtimes. The application mode is reset to the default mode at the end of each Apex test run.

**Signature**

```java
public static Void setReadOnlyApplicationMode(Boolean applicationMode)
```

**Parameters**

*applicationMode*

Type: Boolean

**Return Value**

Type: Void

Usage

Also see the `getApplicationReadWriteMode()` System method.

Do not use `setReadOnlyApplicationMode` for purposes unrelated to Read-Only Mode testing, such as simulating DML exceptions.

Example

The following example sets the application mode to read-only and attempts to insert a new account record, which results in the exception. It then resets the application mode and performs a successful insert.

```java
@isTest
private class ApplicationReadOnlyModeTestClass {
    public static testmethod void test() {
        // Create a test account that is used for querying later.
        Account testAccount = new Account(Name = 'TestAccount');
        insert testAccount;

        // Set the application read only mode.
        Test.setReadOnlyApplicationMode(true);

        // Verify that the application is in read-only mode.
        System.assertEquals( ApplicationReadWriteMode.READ_ONLY, System.getApplicationReadWriteMode() );
    }
}
```
// Create a new account object.
Account testAccount2 = new Account(Name = 'TestAccount2');

try {
    // Get the test account created earlier. Should be successful.
    Account testAccountFromDb = [
        SELECT Id, Name FROM Account WHERE Name = 'TestAccount'
    ];
    System.assertEquals(testAccount.Id, testAccountFromDb.Id);

    // Inserts should result in the InvalidReadOnlyUserDmlException
    // being thrown.
    insert testAccount2;
    System.assertEquals(false, true);
} catch (System.InvalidReadOnlyUserDmlException e) {
    // Expected
}

// Insertion should work after read only application mode gets disabled.
Test.setReadOnlyApplicationMode(false);
insert testAccount2;
Account testAccount2FromDb = [
    SELECT Id, Name FROM Account WHERE Name = 'TestAccount2'
];
System.assertEquals(testAccount2.Id, testAccount2FromDb.Id);
}

startTest()
Marks the point in your test code when your test actually begins. Use this method when you are testing governor limits.

Signature
public static Void startTest()

Return Value
Type: Void

Usage
You can also use this method with stopTest to ensure that all asynchronous calls that come after the startTest method are run before doing any assertions or testing. Each test method is allowed to call this method only once. All of the code before this method should be used to initialize variables, populate data structures, and so on, allowing you to set up everything you need to run your test. Any code that executes after the call to startTest and before stopTest is assigned a new set of governor limits.

stopTest()
Marks the point in your test code when your test ends. Use this method in conjunction with the startTest method.
stopTest()

public static Void stopTest()

Return Value
Type: Void

Usage
Each test method is allowed to call this method only once. Any code that executes after the stopTest method is assigned the original limits that were in effect before startTest was called. All asynchronous calls made after the startTest method are collected by the system. When stopTest is executed, all asynchronous processes are run synchronously.

Note: Asynchronous calls, such as @future or executeBatch, called in a startTest, stopTest block, do not count against your limits for the number of queued jobs.

testInstall(installImplementation, version, isPush)
Tests the implementation of the InstallHandler interface, which is used for specifying a post install script in packages. Tests run as the test initiator in the development environment.

Signature
public static Void testInstall(InstallHandler installImplementation, Version version, Boolean isPush)

Parameters
installImplementation
Type: System.InstallHandler
A class that implements the InstallHandler interface.

version
Type: System.Version
The version number of the existing package installed in the subscriber organization.

isPush
Type: Boolean
(Optional) Specifies whether the upgrade is a push. The default value is false.

Return Value
Type: Void

Usage
This method throws a run-time exception if the test install fails.
Example

```java
@isTest static void test() {
    PostInstallClass postinstall = 
        new PostInstallClass();
    Test.testInstall(postinstall, 
        new Version(1,0));
}
```

testSandboxPostCopyScript(script, organizationId, sandboxId, sandboxName)

Tests the implementation of the SandboxPostCopy Interface, which is used for specifying a script to run at the completion of a Sandbox copy. Tests run as the test initiator in the development environment.

Signature

```java
public static void testSandboxPostCopyScript(System.SandboxPostCopy script, Id organizationId, Id sandboxId, String sandboxName)
```

Parameters

- **script**
  - Type: `System.SandboxPostCopy`
  - A class that implements the SandboxPostCopy interface.

- **organizationId**
  - Type: `Id`
  - The sandbox organization ID

- **sandboxId**
  - Type: `Id`
  - The sandbox ID to be provided to the SandboxPostCopy script.

- **sandboxName**
  - Type: `String`
  - The sandbox name to be provided to the SandboxPostCopy script.

Return Value

Type: `void`

Usage

This method throws a run-time exception if the test install fails.

**Note:** Salesforce recommends that you use the `testSandboxPostCopyScript(script, organizationId, sandboxId, sandboxName, isRunAsAutoProcUser)` overload instead of this method. When `isRunAsAutoProcUser` is true, the SandboxPostCopy script is tested with the same user access permissions as used by post-copy tasks during sandbox creation. Using the same permissions enables the test to better simulate the actual usage of the class, and to uncover potential issues.
Example

See SandboxPostCopy Example Implementation

testSandboxPostCopyScript(script, organizationId, sandboxId, sandboxName, RunAsAutoProcUser)
Tests the implementation of the SandboxPostCopy Interface, which is used for specifying a script to run at the completion of a Sandbox copy. When RunAsAutoProcUser is true, tests run as Automated Process user in the development environment.

Signature

public static void testSandboxPostCopyScript(System.SandboxPostCopy script, Id organizationId, Id sandboxId, String sandboxName, Boolean RunAsAutoProcUser)

Parameters

script
Type: System.SandboxPostCopy
A class that implements the SandboxPostCopy interface.

organizationId
Type: Id
The sandbox organization ID.

sandboxId
Type: Id
The sandbox ID to be provided to the SandboxPostCopy script.

sandboxName
Type: String
The sandbox name to be provided to the SandboxPostCopy script.

RunAsAutoProcUser
Type: Boolean
When true, the SandboxPostCopy script is tested with the same user access permissions as used by post-copy tasks during sandbox creation. Using the same permissions enables the test to better simulate the actual usage of the class, and to uncover potential issues.
When false, the test runs as the test initiator. This option can alter the permissions with which the script is tested, such as the ability to access objects and features.

Return Value

Type: void

Usage

This method throws a run-time exception if the test install fails.
Example
See SandboxPostCopy Example Implementation

testUninstall(uninstallImplementation)
Tests the implementation of the UninstallHandler interface, which is used for specifying an uninstall script in packages. Tests run as the test initiator in the development environment.

Signature
public static Void testUninstall(UninstallHandler uninstallImplementation)

Parameters
uninstallImplementation
Type: System.UninstallHandler
A class that implements the UninstallHandler interface.

Return Value
Type: Void

Usage
This method throws a run-time exception if the test uninstall fails.

Example
@isTest static void test() {
    UninstallClass uninstall =
    new UninstallClass();
    Test.testUninstall(uninstall);
}

Time Class
Contains methods for the Time primitive data type.

Namespace
System

Usage
For more information on time, see Time Data Type.

Time Methods
The following are methods for Time.
IN THIS SECTION:

- `addHours(additionalHours)`
  Adds the specified number of hours to a Time.
- `addMilliseconds(additionalMilliseconds)`
  Adds the specified number of milliseconds to a Time.
- `addMinutes(additionalMinutes)`
  Adds the specified number of minutes to a Time.
- `addSeconds(additionalSeconds)`
  Adds the specified number of seconds to a Time.
- `hour()`
  Returns the hour component of a Time.
- `millisecond()`
  Returns the millisecond component of a Time.
- `minute()`
  Returns the minute component of a Time.
- `newInstance(hour, minutes, seconds, milliseconds)`
  Constructs a Time from Integer representations of the specified hour, minutes, seconds, and milliseconds. (UTC is assumed.)
- `second()`
  Returns the second component of a Time.

### addHours (additionalHours)

Adds the specified number of hours to a Time.

**Signature**

```java
public Time addHours(Integer additionalHours)
```

**Parameters**

- `additionalHours`  
  Type: `Integer`

**Return Value**

Type: `Time`

**Example**

```java
Time myTime = Time.newInstance(1, 2, 3, 4);
Time expected = Time.newInstance(4, 2, 3, 4);
System.assertEquals(expected, myTime.addHours(3));
```

### addMilliseconds (additionalMilliseconds)

Adds the specified number of milliseconds to a Time.

```java
public Time addMilliseconds(Integer additionalMilliseconds)
```
Signature
public Time addMilliseconds(Integer additionalMilliseconds)

Parameters
additionalMilliseconds
Type: Integer

Return Value
Type: Time

Example
Time myTime = Time.newInstance(1, 2, 3, 0);
Time expected = Time.newInstance(1, 2, 4, 400);
System.assertEquals(expected, myTime.addMilliseconds(1400));

addMinutes(additionalMinutes)
Adds the specified number of minutes to a Time.

Signature
public Time addMinutes(Integer additionalMinutes)

Parameters
additionalMinutes
Type: Integer

Return Value
Type: Time

Example
Time myTime = Time.newInstance(18, 30, 2, 20);
Integer myMinutes = myTime.minute();
myMinutes = myMinutes + 5;
System.assertEquals(myMinutes, 35);

addSeconds(additionalSeconds)
Adds the specified number of seconds to a Time.

Signature
public Time addSeconds(Integer additionalSeconds)
Parameters

`additionalSeconds`
Type: Integer

Return Value
Type: Time

Example

```java
Time myTime = Time.newInstance(1, 2, 55, 0);
Time expected = Time.newInstance(1, 3, 5, 0);
System.assertEquals(expected, myTime.addSeconds(10));
```

`hour()`

Returns the hour component of a Time.

Signature

```java
public Integer hour()
```

Return Value
Type: Integer

Example

```java
Time myTime = Time.newInstance(18, 30, 2, 20);
myTime = myTime.addHours(2);
Integer myHour = myTime.hour();
System.assertEquals(myHour, 20);
```

`millisecond()`

Returns the millisecond component of a Time.

Signature

```java
public Integer millisecond()
```

Return Value
Type: Integer

Example

```java
Time myTime = Time.newInstance(3, 14, 15, 926);
System.assertEquals(926, myTime.millisecond());
```
minute()  
Returns the minute component of a Time.  

Signature  
public Integer minute()  

Return Value  
Type: Integer  

Example  
Time myTime = Time.newInstance(3, 14, 15, 926);  
System.assertEquals(14, myTime.minute());  

newInstance(hour, minutes, seconds, milliseconds)  
Constructs a Time from Integer representations of the specified hour, minutes, seconds, and milliseconds. (UTC is assumed.)  

Signature  
public static Time newInstance(Integer hour, Integer minutes, Integer seconds, Integer milliseconds)  

Parameters  
hour  
Type: Integer  

minutes  
Type: Integer  

seconds  
Type: Integer  

milliseconds  
Type: Integer  

Return Value  
Type: Time  

Example  
The following example creates a time of 18:30:2:20 (UTC).  
Time myTime =  
Time.newInstance(18, 30, 2, 20);
second()
Returns the second component of a Time.

Signature
public Integer second()

Return Value
Type: Integer

Example
Time myTime = Time.newInstance(3, 14, 15, 926);
System.assertEquals(15, myTime.second());

TimeZone Class
Represents a time zone. Contains methods for creating a new time zone and obtaining time zone properties, such as the time zone ID, offset, and display name.

Namespace
System

Usage
You can use the methods in this class to get properties of a time zone, such as the properties of the time zone returned by UserInfo.getTimeZone, or the time zone returned by getTimeZone of this class.

Example
This example shows how to get properties of the current user's time zone and displays them to the debug log.

TimeZone tz = UserInfo.getTimeZone();
System.debug('Display name: ' + tz.getDisplayName());
System.debug('ID: ' + tz.getID());
// During daylight saving time for the America/Los_Angeles time zone
System.debug('Offset: ' + tz.getOffset(DateTime.newInstance(2012,10,23,12,0,0)));
// Not during daylight saving time for the America/Los_Angeles time zone
System.debug('Offset: ' + tz.getOffset(DateTime.newInstance(2012,11,23,12,0,0)));
System.debug('String format: ' + tz.toString());

The output of this sample varies based on the user's time zone. This is an example output if the user's time zone is America/Los_Angeles. For this time zone, daylight saving time is -7 hours from GMT (-25200000 milliseconds) and standard time is -8 hours from GMT (-28800000 milliseconds).

Display name: Pacific Standard Time
ID: America/Los_Angeles
Offset: -25200000
This second example shows how to create a time zone for the New York time zone and get the offset of this time zone to the GMT time zone. The example uses two dates to get the offset from. One date is before DST (Daylight Saving Time), and one is after DST. In 2000, DST ended on Sunday, October 29 for the New York time zone. Because the date occurs after DST ends, the offset on the first date is –5 hours to GMT. In 2012, DST ended on Sunday, November 4. Because the date is within DST, the offset on the second date is –4 hours.

```java
// Get the New York time zone
Timezone tz = Timezone.getTimeZone('America/New_York');

// Create a date before the 2007 shift of DST into November
DateTime dtpre = DateTime.newInstanceGMT(2000, 11, 1, 0, 0, 0);
system.debug(tz.getOffset(dtpre)); // -18000000 (= -5 hours = EST)

// Create a date after the 2007 shift of DST into November
DateTime dtpost = DateTime.newInstanceGMT(2012, 11, 1, 0, 0, 0);
system.debug(tz.getOffset(dtpost)); // -14400000 (= -4 hours = EDT)
```

This next example is similar to the previous one except that it gets the offset around the boundary of DST. In 2014, DST ended on Sunday, November 2 at 2:00 AM local time for the New York time zone. The first offset is obtained right before DST ends, and the second offset is obtained right after DST ends. The dates are created by using the `DateTime.newInstanceGMT` method. This method expects the passed-in date values to be based on the GMT time zone.

```java
// Get the New York time zone
Timezone tz = Timezone.getTimeZone('America/New_York');

// Before DST ends
DateTime dtpre = DateTime.newInstanceGMT(2014, 11, 2, 5, 59, 59); // 1:59:59AM local EDT
system.debug(tz.getOffset(dtpre)); // -14400000 (= -4 hours = still on DST)

// After DST ends
DateTime dtpost = DateTime.newInstanceGMT(2014, 11, 2, 6, 0, 0); // 1:00:00AM local EST
system.debug(tz.getOffset(dtpost)); // -18000000 (= -5 hours = back one hour)
```

**TimeZone Methods**

The following are methods for `TimeZone`.

**IN THIS SECTION:**
- `getDisplayName()`
  - Returns this time zone’s display name.
- `getID()`
  - Returns this time zone’s ID.
- `getOffset(date)`
  - Returns the time zone offset, in milliseconds, of the specified date to the GMT time zone.
- `getTimeZone(timeZoneIdString)`
  - Returns the time zone corresponding to the specified time zone ID.
- `toString()`
  - Returns the string representation of this time zone.
**getDisplayName()**
Returns this time zone's display name.

**Signature**
```
public String getDisplayName()
```

**Return Value**
Type: String

**Versioned Behavior Changes**
In API version 45.0 and later, getDisplayName displays Daylight Savings Time appropriately when daylight savings are in effect. For example, British Summer Time is displayed for Europe/London and Pacific Daylight Time for America/Los_Angeles.

**getID()**
Returns this time zone's ID.

**Signature**
```
public String getID()
```

**Return Value**
Type: String

**getOffset(date)**
Returns the time zone offset, in milliseconds, of the specified date to the GMT time zone.

**Signature**
```
public Integer getOffset(Datetime date)
```

**Parameters**
```

date
    Type: Datetime
    The date argument is the date and time to evaluate.
```

**Return Value**
Type: Integer

**Usage**

*Note:* The returned offset is adjusted for daylight saving time if the date argument falls within daylight saving time for this time zone.
getTimeZone(timeZoneIdString)
Returns the time zone corresponding to the specified time zone ID.

Signature
public static TimeZone getTimeZone(String timeZoneIdString)

Parameters
timeZoneIdString
Type: String
The time zone values you can use for the Id argument are any valid time zone values that the Java TimeZone class supports.

Return Value
Type: TimeZone

Example
TimeZone tz = TimeZone.getTimeZone('America/Los_Angeles');
String tzName = tz.getDisplayName();
System.assert(tzName.equals('(GMT-08:00) Pacific Standard Time (America/Los_Angeles)') ||
        tzName.equals('(GMT-07:00) Pacific Daylight Time (America/Los_Angeles)'));

toString()
Returns the string representation of this time zone.

Signature
public String toString()

Return Value
Type: String

Trigger Class
Use the Trigger class to access run-time context information in a trigger, such as the type of trigger or the list of sObject records that the trigger operates on.

Namespace
System

Trigger Context Variables
The Trigger class provides the following context variables.
<table>
<thead>
<tr>
<th><strong>Variable</strong></th>
<th><strong>Usage</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><code>isExecuting</code></td>
<td>Returns true if the current context for the Apex code is a trigger, not a Visualforce page, a Web service, or an <code>executeAnonymous()</code> API call.</td>
</tr>
<tr>
<td><code>isInsert</code></td>
<td>Returns <code>true</code> if this trigger was fired due to an insert operation, from the Salesforce user interface, Apex, or the API.</td>
</tr>
<tr>
<td><code>isUpdate</code></td>
<td>Returns <code>true</code> if this trigger was fired due to an update operation, from the Salesforce user interface, Apex, or the API.</td>
</tr>
<tr>
<td><code>isDelete</code></td>
<td>Returns <code>true</code> if this trigger was fired due to a delete operation, from the Salesforce user interface, Apex, or the API.</td>
</tr>
<tr>
<td><code>isBefore</code></td>
<td>Returns <code>true</code> if this trigger was fired before any record was saved.</td>
</tr>
<tr>
<td><code>isAfter</code></td>
<td>Returns <code>true</code> if this trigger was fired after all records were saved.</td>
</tr>
<tr>
<td><code>isUndelete</code></td>
<td>Returns <code>true</code> if this trigger was fired after a record is recovered from the Recycle Bin. This recovery can occur after an undelete operation from the Salesforce user interface, Apex, or the API.</td>
</tr>
<tr>
<td><code>new</code></td>
<td>Returns a list of the new versions of the sObject records. This sObject list is only available in <code>insert</code>, <code>update</code>, and <code>undelete</code> triggers, and the records can only be modified in <code>before</code> triggers.</td>
</tr>
<tr>
<td><code>newMap</code></td>
<td>A map of IDs to the new versions of the sObject records. This map is only available in <code>before update</code>, <code>after insert</code>, <code>after update</code>, and <code>after undelete</code> triggers.</td>
</tr>
<tr>
<td><code>old</code></td>
<td>Returns a list of the old versions of the sObject records. This sObject list is only available in <code>update</code> and <code>delete</code> triggers.</td>
</tr>
<tr>
<td><code>oldMap</code></td>
<td>A map of IDs to the old versions of the sObject records. This map is only available in <code>update</code> and <code>delete</code> triggers.</td>
</tr>
<tr>
<td><code>operationType</code></td>
<td>Returns an enum of type <code>System.TriggerOperation</code> corresponding to the current operation. Possible values of the <code>System.TriggerOperation</code> enum are: <code>BEFORE_INSERT</code>, <code>BEFORE_UPDATE</code>, <code>BEFORE_DELETE</code>, <code>AFTER_INSERT</code>, <code>AFTER_UPDATE</code>, <code>AFTER_DELETE</code>, and <code>AFTER_UNDELETE</code>. If you vary your programming logic based on different trigger types, consider using the <code>switch</code> statement with different permutations of unique trigger execution enum states.</td>
</tr>
<tr>
<td><code>size</code></td>
<td>The total number of records in a trigger invocation, both old and new.</td>
</tr>
</tbody>
</table>

**Note:** The record firing a trigger can include an invalid field value, such as a formula that divides by zero. In this case, the field value is set to `null` in these variables:

- `new`
- `newMap`
- `old`
- `oldMap`
### Example

For example, in this simple trigger, `Trigger.new` is a list of sObjects and can be iterated over in a `for` loop. It can also be used as a bind variable in the `IN` clause of a SOQL query.

```apex
Trigger simpleTrigger on Account (after insert) {
    for (Account a : Trigger.new) {
        // Iterate over each sObject
    }

    // This single query finds every contact that is associated with any of the
    // triggering accounts. Note that although Trigger.new is a collection of
    // records, when used as a bind variable in a SOQL query, Apex automatically
    // transforms the list of records into a list of corresponding Ids.
    Contact[] cons = [SELECT LastName FROM Contact
                     WHERE AccountId IN :Trigger.new];
}
```

This trigger uses Boolean context variables like `Trigger.isBefore` and `Trigger.isDelete` to define code that only executes for specific trigger conditions:

```apex
trigger myAccountTrigger on Account(before delete, before insert, before update,
                                    after delete, after insert, after update) {
    if (Trigger.isBefore) {
        if (Trigger.isDelete) {
            // In a before delete trigger, the trigger accesses the records that will be
            // deleted with the Trigger.old list.
            for (Account a : Trigger.old) {
                if (a.name != 'okToDelete') {
                    a.addError('You can\'t delete this record!');
                }
            }
        } else {
            // In before insert or before update triggers, the trigger accesses the new records
            // with the Trigger.new list.
            for (Account a : Trigger.new) {
                if (a.name == 'bad') {
                    a.name.addError('Bad name');
                }
            }
        }
    } else {
        if (Trigger.isInsert) {
            List<Contact> contacts = new List<Contact>();

            for (Account a : Trigger.new) {
                System.assertEquals('xxx', a.accountNumber);
                System.assertEquals('industry', a.industry);
                System.assertEquals(100, a.numberofemployees);
                System.assertEquals(100.0, a.annualrevenue);
                a.accountNumber = 'yyy';
            }
        }
    }
}
```

// If the trigger is not a before trigger, it must be an after trigger.
for (Account a : Trigger.new) {
    if(a.Name == 'makeContact') {
        contacts.add(new Contact (LastName = a.Name, AccountId = a.Id));
    }
    insert contacts;
}
}}

TriggerOperation Enum

System.TriggerOperation enum values are associated with trigger events.

Enum Values

The following are the values of the System.TriggerOperation enum:

- AFTER_DELETE
- AFTER_INSERT
- AFTER_UNDELETE
- AFTER_UPDATE
- BEFORE_DELETE
- BEFORE_INSERT
- BEFORE_UPDATE

Type Class

Contains methods for getting the Apex type that corresponds to an Apex class and for instantiating new types.

Namespace

System

Usage

Use the forName methods to retrieve the type of an Apex class, which can be a built-in or a user-defined class. You can use these methods to retrieve the type of public and global classes, and not private classes even if the context user has access. Also, use the newInstance method if you want to instantiate a Type that implements an interface and call its methods while letting someone else, such as a subscriber of your package, provide the methods' implementations.

Note: A call to Type.forName() can cause the class to be compiled.

Example: Instantiating a Type Based on Its Name

The following sample shows how to use the Type methods to instantiate a Type based on its name. A typical application of this scenario is when a package subscriber provides a custom implementation of an interface that is part of an installed package. The package can
get the name of the class that implements the interface through a custom setting in the subscriber’s org. The package can then instantiate the type that corresponds to this class name and invoke the methods that the subscriber implemented.

In this sample, Vehicle represents the interface that the VehicleImpl class implements. The last class contains the code sample that invokes the methods implemented in VehicleImpl.

This is the Vehicle interface.

```apex
global interface Vehicle {
    Long getMaxSpeed();
    String getType();
}
```

This is the implementation of the Vehicle interface.

```apex
global class VehicleImpl implements Vehicle {
    global Long getMaxSpeed() { return 100; }
    global String getType() { return 'Sedan'; }
}
```

The method in this class gets the name of the class that implements the Vehicle interface through a custom setting value. It then instantiates this class by getting the corresponding type and calling the newInstance method. Next, it invokes the methods implemented in VehicleImpl. This sample requires that you create a public list custom setting named CustomImplementation with a text field named className. Create one record for this custom setting with a data set name of Vehicle and a class name value of VehicleImpl.

```apex
public class CustomerImplInvocationClass {

    public static void invokeCustomImpl() {
        // Get the class name from a custom setting.
        // This class implements the Vehicle interface.
        CustomImplementation__c cs = CustomImplementation__c.getInstance('Vehicle');

        // Get the Type corresponding to the class name
        Type t = Type.forName(cs.className__c);

        // Instantiate the type.
        // The type of the instantiated object
        // is the interface.
        Vehicle v = (Vehicle)t.newInstance();

        // Call the methods that have a custom implementation
        System.debug('Max speed: ' + v.getMaxSpeed());
        System.debug('Vehicle type: ' + v.getType());
    }
}
```

Class Property

The class property returns the System.Type of the type it is called on. It’s exposed on all Apex built-in types including primitive data types and collections, sObject types, and user-defined classes. This property can be used instead of forName methods.

Call this property on the type name. For example:

```apex
System.Type t = Integer.class;
```
You can use this property for the second argument of JSON.deserialize, deserializeStrict, JSONParser.readValueAs, and readValueAsStrict methods to get the type of the object to deserialize. For example:

```java
Decimal n = (Decimal)JSON.deserialize('100.1', Decimal.class);
```

### Type Methods

The following are methods for `Type`.

**IN THIS SECTION:**

- `equals(typeToCompare)`: Returns `true` if the specified type is equal to the current type; otherwise, returns `false`.
- `forName(fullyQualifiedName)`: Returns the type that corresponds to the specified fully qualified class name.
- `forName(namespace, name)`: Returns the type that corresponds to the specified namespace and class name.
- `getName()`: Returns the name of the current type.
- `hashCode()`: Returns a hash code value for the current type.
- `isAssignableFrom(sourceType)`: Returns `true` if an object reference of the specified type can be assigned from the child type; otherwise, returns `false`.
- `newInstance()`: Creates an instance of the current type and returns this new instance.
- `toString()`: Returns a string representation of the current type, which is the type name.

**equals(typeToCompare)**

Returns `true` if the specified type is equal to the current type; otherwise, returns `false`.

**Signature**

```java
public Boolean equals(Object typeToCompare)
```

**Parameters**

- `typeToCompare`  
  Type: `Object`  
  The type to compare with the current type.

**Return Value**

Type: `Boolean`
Example

```java
Type t1 = Account.class;
Type t2 = Type.forName('Account');
System.assert(t1.equals(t2));
```

**forName(fullyQualifiedName)**

Returns the type that corresponds to the specified fully qualified class name.

**Signature**

```java
public static System.Type forName(String fullyQualifiedName)
```

**Parameters**

`fullyQualifiedName`  
Type: String  
The fully qualified name of the class to get the type of. The fully qualified class name contains the namespace name, for example, `MyNamespace.ClassName`.

**Return Value**

Type: `System.Type`

**Usage**

⚠️ **Note:**

- This method returns `null` if called outside a managed package to get the type of a non-global class in a managed package. This is because the non-global class isn’t visible outside the managed package. For Apex saved using Salesforce API version 27.0 and earlier, this method does return the corresponding class type for the non-global managed package class.
- When called from an installed managed package to get the name of a local type in an organization with no defined namespace, the `forName(fullyQualifiedName)` method returns `null`. Instead, use the `forName(namespace, name)` method and specify an empty string or `null` for the namespace argument.
- A call to `Type.forName()` can cause the class to be compiled.

**forName(namespace, name)**

Returns the type that corresponds to the specified namespace and class name.

**Signature**

```java
public static System.Type forName(String namespace, String name)
```

**Parameters**

`namespace`  
Type: String  
The namespace of the class. If the class doesn’t have a namespace, set the `namespace` argument to `null` or an empty string.
name
Type: String
The name of the class.

Return Value
Type: System.Type

Usage
Note:
• This method returns null if called outside a managed package to get the type of a non-global class in a managed package. This is because the non-global class isn’t visible outside the managed package. For Apex saved using Salesforce API version 27.0 and earlier, this method does return the corresponding class type for the non-global managed package class.
• Use this method instead of forName(fullyQualifiedName) if it’s called from a managed package installed in an organization with no defined namespace. To get the name of a local type, set the namespace argument to an empty string or null. For example, Type t = Type.forName('', 'ClassName');
• A call to Type.forName() can cause the class to be compiled.

Example
This example shows how to get the type that corresponds to the ClassName class and the MyNamespace namespace.

Type myType =
    Type.forName('MyNamespace', 'ClassName');

getName()
Returns the name of the current type.

Signature
public String getName()

Return Value
Type: String

Example
This example shows how to get a Type’s name. It first obtains a Type by calling forName, then calls getName on the Type object.

Type t =
    Type.forName('MyClassName');

String typeName =
    t.getName();
System.assertEquals('MyClassName',
    typeName);
hashCode()
Returns a hash code value for the current type.

Signature
public Integer hashCode()

Return Value
Type: Integer

Usage
The returned hash code value corresponds to the type name hash code that String.hashCode returns.

isAssignableFrom(sourceType)
Returns true if an object reference of the specified type can be assigned from the child type; otherwise, returns false.

Signature
public Boolean isAssignableFrom(Type sourceType)

Parameters
sourceType
The type of the object with which you are checking compatibility.

Return Value
Type: Boolean
The method returns true when the method is invoked as parentType.isAssignableFrom(childType). When invoked in any of the following ways, the method returns false:
- childType.isAssignableFrom(parentType)
- typeA.isAssignableFrom(TypeB) where TypeB is a sibling of TypeA
- typeA.isAssignableFrom(TypeB) where TypeB and TypeA are unrelated

Note: A childType is the child of a parentType when it implements an interface, extends a virtual or abstract class, or is the same System.Type as the parentType.

Usage
Unlike the instanceof operator, this method allows you to check type compatibility without having to create a class instance. This method eliminates static compile-time dependencies that instanceof requires.

The following code demonstrates how a typical ISV customer can use isAssignableFrom() to check compatibility between a customer-defined type (customerProvidedPluginType) and a valid plugin type.

//Scenario: Managed package code loading a “plugin” class that implements a managed interface; the implementation done outside of the package
String pluginNameStr = Config__c.getInstance().PluginApexType__c;
Type customerProvidedPluginType = Type.forName(pluginNameStr);
Type pluginInterface = ManagedPluginInterface.class;

// Constructors may have side-effects, including potentially unsafe DML/callouts.
// We want to make sure the class is really designed to be a valid plugin before we
// instantiate it
Boolean validPlugin = pluginInterface.isAssignableFrom(customerProvidedPluginType); //
// validate that it implements the right interface

if(!validPlugin){
    throw new SecurityException('Cannot create instance of '+customerProvidedPluginType+'.
Does not implement ManagedPluginInterface');
} else{
    return Type.newInstance(validPlugin);
}

Example

The following code snippet first defines sibling classes A and B that both implement the Callable interface and an unrelated class C.
Then, it explores several type comparisons using isAssignableFrom().

//Define classes A, B, and C

global class A implements Database.Batchable<String>, Callable {
    global Iterable<String> start(Database.BatchableContext context) { return null; }
    global void execute(Database.BatchableContext context, String[] scope) { }
    global void finish(Database.BatchableContext context) { }
    global Object call(String action, Map<String, Object> args) { return null; }
}

global class B implements Callable {
    global Object call(String action, Map<String, Object> args) { return null; }
}

global class C {
}

Type listOfStrings = Type.forName('List<String>');
Type listOfIntegers = Type.forName('List<Integer>');
boolean flagListTypes = listOfIntegers.isAssignableFrom(listOfStrings); // false

//Examples with stringType and idType
Type stringType = Type.forName('String');
Type idType = Type.forName('Id');
boolean isId_assignableFromString = idType.isAssignableFrom(stringType); // true
//isAssignableFrom respects that String can be assigned to Id without an explicit cast

//Examples with typeA, typeB, and typeC
Type typeA = Type.forName('A');
Type typeB = Type.forName('B');
Type typeC = Type.forName('C');
boolean isTypeB_ofTypeA = typeB.isAssignableFrom(typeA); // false - siblings
boolean isTypeA_ofTypeC = typeA.isAssignableFrom(typeC); // false - unrelated types
boolean isTypeA_ofTypeA = typeA.isAssignableFrom(typeA); // true - identity

//Examples with callableType and batchableType
Type callableType = Type.forName('Callable');
Type batchableType = Type.forName('Database.Batchable');
boolean isTypeA_Callable = callableType.isAssignableFrom(typeA); // true - type A is a child of Callable type
boolean isTypeA_Batchable = batchableType.isAssignableFrom(typeA); // true - type A is a child of Batchable type
boolean isCallableOfTypeA = typeA.isAssignableFrom(callableType); // false - Callable type is not a child of type A
boolean isBatchableOfTypeA = typeA.isAssignableFrom(batchableType); // false - Batchable type is not a child of type A

newInstance()

Creates an instance of the current type and returns this new instance.

Signature

public Object newInstance()

Return Value

Type: Object

Usage

Because newInstance returns the generic object type, you should cast the return value to the type of the variable that will hold this value.

This method enables you to instantiate a Type that implements an interface and call its methods while letting someone else provide the methods’ implementation. For example, a package developer can provide an interface that a subscriber who installs the package can implement. The code in the package calls the subscriber’s implementation of the interface methods by instantiating the subscriber’s Type.

Example

This example shows how to create an instance of a Type. It first gets a Type by calling forName with the name of a class (ShapeImpl), then calls newInstance on this Type object. The newObj instance is declared with the interface type (Shape) that the ShapeImpl class implements. The return value of the newInstance method is cast to the Shape type.

Type t =
    Type.forName('ShapeImpl');

Shape newObj =
    (Shape)t.newInstance();

toString()

Returns a string representation of the current type, which is the type name.
Signature

```
public String toString()
```

Return Value

Type: String

Usage

This method returns the same value as `getName`. `String.valueOf` and `System.debug` use this method to convert their `Type` argument into a String.

Example

This example calls `toString` on the `Type` corresponding to a list of Integers.

```
Type t = List<Integer>.class;
String s = t.toString();
System.assertEquals('List<Integer>', s);
```

UninstallHandler Interface

Enables custom code to run after a managed package is uninstalled.

Namespace

`System`

Usage

App developers can implement this interface to specify Apex code that runs automatically after a subscriber uninstalls a managed package. This makes it possible to perform cleanup and notification tasks based on details of the subscriber’s organization.

The uninstall script is subject to default governor limits. It runs as a special system user that represents your package, so all operations performed by the script will appear to be done by your package. You can access this user by using `UserInfo`. You will only see this user at runtime, not while running tests.

If the script fails, the uninstall continues but none of the changes performed by the script are committed. Any errors in the script are emailed to the user specified in the `Notify on Apex Error` field of the package. If no user is specified, the uninstall details will be unavailable.

The uninstall script has the following restrictions. You can’t use it to initiate batch, scheduled, and future jobs, to access Session IDs, or to perform callouts.

The `UninstallHandler` interface has a single method called `onUninstall`, which specifies the actions to be performed on uninstall.

```
global interface UninstallHandler {
    void onUninstall(UninstallContext context);
}
```

The `onUninstall` method takes a context object as its argument, which provides the following information:

- The org ID of the organization in which the uninstall takes place.
• The user ID of the user who initiated the uninstall.

The context argument is an object whose type is the UninstallContext interface. This interface is automatically implemented by the system. The following definition of the UninstallContext interface shows the methods you can call on the context argument.

global interface UninstallContext {
  ID organizationId();
  ID uninstallerId();
}

IN THIS SECTION:
  UninstallHandler Methods
  UninstallHandler Example Implementation

UninstallHandler Methods

The following are methods for UninstallHandler.

IN THIS SECTION:
  onUninstall(context)
    Specifies the actions to be performed on uninstall.

onUninstall(context)

Specifies the actions to be performed on uninstall.

Signature

public Void onUninstall(UninstallContext context)

Parameters

  context
    Type: UninstallContext

Return Value

Type: Void

UninstallHandler Example Implementation

Example of an Uninstall Script

This sample uninstall script performs the following actions on package uninstall.

• Inserts an entry in the feed describing which user did the uninstall and in which organization
• Creates and sends an email message confirming the uninstall to that user

```apex
global class UninstallClass implements UninstallHandler {
    global void onUninstall(UninstallContext ctx) {
        FeedItem feedPost = new FeedItem();
        feedPost.parentId = ctx.uninstallerID();
        feedPost.body = 'Thank you for using our application!';
        insert feedPost;

        User u = [Select Id, Email from User where Id =:ctx.uninstallerID()];
        String toAddress= u.Email;
        String[] toAddresses = new String[] {toAddress};
        Messaging.SingleEmailMessage mail = new Messaging.SingleEmailMessage();
        mail.setToAddresses(toAddresses);
        mail.setReplyTo('support@package.dev');
        mail.setSenderDisplayName('My Package Support');
        mail.setSubject('Package uninstall successful');
        mail.setPlainTextBody('Thanks for uninstalling the package.');
        Messaging.sendEmail(new Messaging.Email[] { mail });
    }
}
```

You can test an uninstall script using the `testUninstall` method of the `Test` class. This method takes as its argument a class that implements the `UninstallHandler` interface.

This sample shows how to test an uninstall script implemented in the `UninstallClass` Apex class.

```apex
@isTest
static void testUninstallScript() {
    Id UninstallerId = UserInfo.getUserId();
    List<FeedItem> feedPostsBefore =
        [SELECT Id FROM FeedItem WHERE parentId=:UninstallerId AND CreatedDate=TODAY];
    Test.testUninstall(new UninstallClass());
    List<FeedItem> feedPostsAfter =
        [SELECT Id FROM FeedItem WHERE parentId=:UninstallerId AND CreatedDate=TODAY];
    System.assertEquals(feedPostsBefore.size() + 1, feedPostsAfter.size(),
                        'Post to uninstaller failed.');
}
```

**URL Class**

Represents a uniform resource locator (URL) and provides access to parts of the URL. Enables access to the base URL used to access your Salesforce org.

**Namespace**

System
Usage

Use the methods of the `System.URL` class to create links to objects in your organization. Such objects can be files, images, logos, or records that you want to include in external emails, in activities, or in Chatter posts. For example, you can create a link to a file uploaded as an attachment to a Chatter post by concatenating the Salesforce base URL with the file ID:

```java
// Get a file uploaded through Chatter.
ContentDocument doc = [SELECT Id FROM ContentDocument WHERE Title = 'myfile'];
// Create a link to the file.
String fullFileURL = URL.getOrgDomainURL().toExternalForm() + '/' + doc.id;
system.debug(fullFileURL);
```

The following example creates a link to a Salesforce record. The full URL is created by concatenating the Salesforce base URL with the record ID.

```java
Account acct = [SELECT Id FROM Account WHERE Name = 'Acme' LIMIT 1];
String fullRecordURL = URL.getOrgDomainURL().toExternalForm() + '/' + acct.Id;
```

Example

In this example, the base URL and the full request URL of the current Salesforce server instance are retrieved. Next, a URL pointing to a specific account object is created. Finally, components of the base and full URL are obtained. This example prints out all the results to the debug log output.

```java
// Create a new account called Acme that we will create a link for later.
Account myAccount = new Account(Name='Acme');
insert myAccount;
// Get the base URL.
String sfdcBaseURL = URL.getOrgDomainURL().toExternalForm();
System.debug('Base URL: ' + sfdcBaseURL);
// Get the URL for the current request.
String currentRequestURL = URL.getCurrentRequestUrl().toExternalForm();
System.debug('Current request URL: ' + currentRequestURL);
// Create the account URL from the base URL.
String accountURL = URL.getOrgDomainURL().toExternalForm() + '/' + myAccount.Id;
System.debug('URL of a particular account: ' + accountURL);
// Get some parts of the base URL.
System.debug('Host: ' + URL.getOrgDomainURL().getHost());
System.debug('Protocol: ' + URL.getOrgDomainURL().getProtocol());
// Get the query string of the current request.
System.debug('Query: ' + URL.getCurrentRequestUrl().getQuery());
```

Versioned Behavior Changes

In API version 41.0 and later, Apex URL objects are represented by the `java.net.URI` type, not the `java.net.URL` type. The API version in which the URL object was instantiated determines the behavior of subsequent method calls to the specific instance.
Salesforce strongly encourages you to use API 41.0 and later versions for fully RFC-compliant URL parsing that includes proper handling of edge cases of complex URL structures. API 41.0 and later versions also enforce that inputs are valid, RFC-compliant URL or URI strings.

IN THIS SECTION:
- URL Constructors
- URL Methods

### URL Constructors

The following are constructors for `URL`.

IN THIS SECTION:
- `Url(spec)`
  - Creates a new instance of the `URL` class using the specified string representation of the URL.
- `Url(context, spec)`
  - Creates a new instance of the `URL` class by parsing the specified spec within the specified context.
- `Url(protocol, host, file)`
  - Creates a new instance of the `URL` class using the specified protocol, host, and file on the host. The default port for the specified protocol is used.
- `Url(protocol, host, port, file)`
  - Creates a new instance of the `URL` class using the specified protocol, host, port, and file on the host.

#### `Url(spec)`

Creates a new instance of the `URL` class using the specified string representation of the URL.

**Signature**

```java
public URL(String spec)
```

**Parameters**

- `spec`
  - Type: `String`
  - The string to parse as a URL.

#### `Url(context, spec)`

Creates a new instance of the `URL` class by parsing the specified spec within the specified context.

**Signature**

```java
public URL(Url context, String spec)
```
Parameters

context
  Type: URL on page 3542
  The context in which to parse the specification.

spec
  Type: String
  The string to parse as a URL.

Usage

The new URL is created from the given context URL and the spec argument as described in RFC2396 "Uniform Resource Identifiers: Generic Syntax":

```
<scheme>://<authority><path>?<query>#<fragment>
```

For more information about the arguments of this constructor, see the corresponding URL(java.net.URL, java.lang.String) constructor for Java.

Url(protocol, host, file)

Creates a new instance of the URL class using the specified protocol, host, and file on the host. The default port for the specified protocol is used.

Signature

```
public Url(String protocol, String host, String file)
```

Parameters

protocol
  Type: String
  The protocol name for this URL.

host
  Type: String
  The host name for this URL.

file
  Type: String
  The file name for this URL.

Url(protocol, host, port, file)

Creates a new instance of the URL class using the specified protocol, host, port, and file on the host.

Signature

```
public Url(String protocol, String host, Integer port, String file)
```
Parameters

**protocol**
Type: *String*
The protocol name for this URL.

**host**
Type: *String*
The host name for this URL.

**port**
Type: *Integer*
The port number for this URL.

**file**
Type: *String*
The file name for this URL.

URL Methods

The following are methods for URL.

IN THIS SECTION:

- `getAuthority()`
  Returns the authority portion of the current URL.

- `getCurrentRequestUrl()`
  Returns the URL of an entire request on a Salesforce instance.

- `getDefaultPort()`
  Returns the default port number of the protocol associated with the current URL.

- `getFile()`
  Returns the file name of the current URL.

- `getFileFieldURL(entityId, fieldName)`
  Returns the download URL for a file attachment.

- `getHost()`
  Returns the host name of the current URL.

- `getOrgDomainUrl()`
  Returns the canonical URL for your org. For example, https://MyDomainName.my.salesforce.com.

- `getPath()`
  Returns the path portion of the current URL.

- `getPort()`
  Returns the port of the current URL.

- `getProtocol()`
  Returns the protocol name of the current URL, such as, https.

- `getQuery()`
  Returns the query portion of the current URL.
getRef()
Returns the anchor of the current URL.

getSalesforceBaseUrl()
In API version 59.0 and later, this method is deprecated and versioned out. Use getOrgDomainUrl() to get the canonical URL for your org or use getCurrentRequestUrl() to get the URL of an entire request on a Salesforce instance. Returns the URL of the current connection to the Salesforce org.

getUserInfo()
Gets the UserInfo portion of the current URL.
sameFile(URLToCompare)
Compares the current URL with the specified URL object, excluding the fragment component.
toExternalForm()
Returns a string representation of the current URL.

getAuthority()
Returns the authority portion of the current URL.

Signature
public String getAuthority()

Return Value
Type: String

getCurrentRequestUrl()
Returns the URL of an entire request on a Salesforce instance.

Signature
public static System.URL getCurrentRequestUrl()

Return Value
Type: System.URL

Usage
An example of a URL for an entire request is https://yourInstance.salesforce.com/apex/myVfPage.apexp.

defaultPort()
Returns the default port number of the protocol associated with the current URL.

Signature
public Integer getDefaultPort()
Return Value
Type: Integer

Usage
Returns -1 if the URL scheme or the stream protocol handler for the URL doesn’t define a default port number.

getFileSize()
Returns the file name of the current URL.

Signature
public String getFile()

Return Value
Type: String

ggetFileFieldURL(entityId, fieldName)
Returns the download URL for a file attachment.

Signature
public static String getFileFieldURL(String entityId, String fieldName)

Parameters
entityId
Type: String
Specifies the ID of the entity that holds the file data.

fieldName
Type: String
Specifies the API name of a file field component, such as AttachmentBody.

Return Value
Type: String

Usage
Example:

Example
String fileURL =
URL.getFileFieldURL( 
getHost()

Returns the host name of the current URL.

Signature

public String getHost()

Return Value

Type: String

getOrgDomainUrl()

Returns the canonical URL for your org. For example, https://MyDomainName.my.salesforce.com.

Signature

public static System.Url getOrgDomainUrl()

Return Value

Type: System.URL

getOrgDomainUrl() always returns the login URL for your org, regardless of context. Use that URL when making API calls to your org.

Usage

Use getOrgDomainUrl() to interact with Salesforce REST and SOAP APIs in Apex code. Get endpoints for User Interface API calls, for creating and customizing picklist value sets and custom fields, and more.

getOrgDomainUrl() can access the domain URL only for the org in which the Apex code is running.

You don’t need a RemoteSiteSetting for your org to interact with the Salesforce APIs using domain URLs retrieved with this method.

Example

This example uses the Salesforce REST API to get organization limit values. For information on limits, see Limits in the REST API Developer Guide.

```java
Http h = new Http();
HttpRequest req = new HttpRequest();
req.setEndpoint(Url.getOrgDomainUrl().toExternalForm() + '/services/data/v44.0/limits');
req.setMethod('GET');
```
req.setHeader('Authorization', 'Bearer ' + UserInfo.getSessionId());
HttpResponse res = h.send(req);

SEE ALSO:
Lightning Aura Components Developer Guide: Making API Calls from Apex
User Interface API Developer Guide: Get Default Values to Clone a Record
User Interface API Developer Guide: Get Values for a Picklist Field
User Interface API Developer Guide: User Inteface API Resources

**getPath()**

Returns the path portion of the current URL.

**Signature**

```java
public String getPath()
```

**Return Value**

Type: String

**getPort()**

Returns the port of the current URL.

**Signature**

```java
public Integer getPort()
```

**Return Value**

Type: Integer

**getProtocol()**

Returns the protocol name of the current URL, such as, https.

**Signature**

```java
public String getProtocol()
```

**Return Value**

Type: String

**getQuery()**

Returns the query portion of the current URL.
**Signature**

```java
public String getQuery()
```

**Return Value**

Type: `String`

**Usage**

Returns `null` if no query portion exists.

**getRef()**

Retruns the anchor of the current URL.

**Signature**

```java
public String getRef()
```

**Return Value**

Type: `String`

**Usage**

Returns `null` if no query portion exists.

**getSalesforceBaseUrl()**

In API version 59.0 and later, this method is deprecated and versioned out. Use `getOrgDomainUrl()` to get the canonical URL for your org or use `getCurrentRequestUrl()` to get the URL of an entire request on a Salesforce instance. Returns the URL of the current connection to the Salesforce org.

**Signature**

```java
public static System.URL getSalesforceBaseUrl()
```

**Return Value**

Type: `System.URL`


**SEE ALSO:**

- `getOrgDomainUrl()`

**getUserInfo()**

Gets the UserInfo portion of the current URL.
Signature
public String getUserInfo()

Return Value
Type: String

Usage
Returns null if no UserInfo portion exists.

**sameFile(URLToCompare)**
Compares the current URL with the specified URL object, excluding the fragment component.

Signature
public Boolean sameFile(System.URL URLToCompare)

Parameters
URLToCompare
  Type: System.URL

Return Value
Type: Boolean
Returns true if both URL objects reference the same remote resource; otherwise, returns false.

Usage
For more information about the syntax of URIs and fragment components, see RFC3986.

toExternalForm()
Returns a string representation of the current URL.

Signature
public String toExternalForm()

Return Value
Type: String

UserInfo Class
Contains methods for obtaining information about the context user.
Namespace

System

UserInfo Methods

The following are methods for UserInfo. All methods are static.

IN THIS SECTION:

- `getDefaultCurrency()`: Returns the context user’s default currency code for multiple currency organizations or the organization’s currency code for single currency organizations.
- `getFirstName()`: Returns the context user’s first name
- `getLanguage()`: Returns the context user’s language
- `getLastName()`: Returns the context user’s last name
- `getLocale()`: Returns the context user’s locale.
- `getName()`: Returns the context user’s full name. The format of the name depends on the language preferences specified for the organization.
- `getOrganizationId()`: Returns the context organization’s ID.
- `getOrganizationName()`: Returns the context organization’s company name.
- `getProfileId()`: Returns the context user’s profile ID.
- `getSessionId()`: Returns the session ID for the current session.
- `getTimeZone()`: Returns the current user’s local time zone.
- `getUiTheme()`: Returns the preferred theme for the current user. Use `getUiThemeDisplayed` to determine the theme actually displayed to the current user.
- `getUiThemeDisplayed()`: Returns the theme being displayed for the current user.
- `getUserEmail()`: Returns the current user’s email address.
- `getUserId()`: Returns the context user’s ID
getUserName()  
Returns the context user's login name.

getUserRoleId()  
Returns the context user's role ID.

getUserType()  
Returns the context user's type.

hasPackageLicense(packageId)  
Returns true if the context user has a license to the managed package via a package license only. Otherwise, returns false.

isCurrentUserLicensed(namespace)  
Returns true if the context user has a license to any managed package denoted by the namespace. Otherwise, returns false.

isCurrentUserLicensedForPackage(packageId)  
Returns true if the context user has a license to the managed package denoted by the package ID. Otherwise, returns false. If the context user has access, it's determined either via the package license or a namespace permission set license for the package namespace.

isMultiCurrencyOrganization()  
Specifies whether the organization uses multiple currencies.

getDefaultCurrency()  
Returns the context user's default currency code for multiple currency organizations or the organization's currency code for single currency organizations.

**Signature**

```
public static String getDefaultCurrency()
```

**Return Value**

Type: String

**Usage**

Note: For Apex saved using Salesforce API version 22.0 or earlier, `getDefaultCurrency` returns null for single currency organizations.

getFirstName()  
Returns the context user's first name.

**Signature**

```
public static String getFirstName()
```

**Return Value**

Type: String
**getLanguage()**

Returns the context user's language

**Signature**

```java
public static String getLanguage()
```

**Return Value**

Type: String

**getLastName()**

Returns the context user's last name

**Signature**

```java
public static String getLastName()
```

**Return Value**

Type: String

**getLocale()**

Returns the context user's locale.

**Signature**

```java
public static String getLocale()
```

**Return Value**

Type: String

**Example**

```java
String result = UserInfo.getLocale();
System.assertEquals('en_US', result);
```

**getName()**

Returns the context user's full name. The format of the name depends on the language preferences specified for the organization.

**Signature**

```java
public static String getName()
```
Return Value
Type: String

Usage
The format is one of the following:

- FirstName LastName
- LastName, FirstName

getOrganizationId()
Returns the context organization’s ID.

Signature
public static String getOrganizationId()

Return Value
Type: String

getOrganizationName()
Returns the context organization’s company name.

Signature
public static String getOrganizationName()

Return Value
Type: String

getProfileId()
Returns the context user’s profile ID.

Signature
public static String getProfileId()

Return Value
Type: String

getSessionId()
Returns the session ID for the current session.
**Signature**

```java
public static String getSessionId()
```

**Return Value**

Type: String

**Usage**

You can use `getSessionId()` both synchronously and asynchronously. In asynchronous Apex (Batch, Future, Queueable, or Scheduled Apex), this method returns the session ID only when the code is run by an active, valid user. When the code is run by an internal user, such as the automated process user or a proxy user, the method returns `null`.

As a best practice, ensure that your code handles both cases: when a session ID is or is not available.

**getTimeZone()**

Returns the current user’s local time zone.

**Signature**

```java
public static System.TimeZone getTimeZone()
```

**Return Value**

Type: System.TimeZone

**Example**

```java
TimeZone tz = UserInfo.getTimeZone();
System.debug('Display name: ' + tz.getDisplayName());
System.debug('ID: ' + tz.getID());
```

**getUiTheme()**

Returns the preferred theme for the current user. Use `getUiThemeDisplayed` to determine the theme actually displayed to the current user.

**Signature**

```java
public static String getUiTheme()
```

**Return Value**

Type: String

The preferred theme for the current user.
Valid values include:

- Theme1—Obsolete Salesforce theme
- Theme2—Salesforce Classic 2005 user interface theme
- Theme3—Salesforce Classic 2010 user interface theme
- Theme4d—Modern "Lightning Experience" Salesforce theme
- Theme4t—Salesforce mobile app theme
- Theme4u—Lightning Console theme
- PortalDefault—Salesforce Customer Portal theme that applies to Customer Portals only and not to Experience Builder sites
- Webstore—AppExchange theme

**getUiThemeDisplayed()**

Returns the theme being displayed for the current user.

**Signature**

```java
public static String getUiThemeDisplayed()
```

**Return Value**

Type: **String**

The theme being displayed for the current user

Valid values include:

- Theme1—Obsolete Salesforce theme
- Theme2—Salesforce Classic 2005 user interface theme
- Theme3—Salesforce Classic 2010 user interface theme
- Theme4d—Modern "Lightning Experience" Salesforce theme
- Theme4t—Salesforce mobile app theme
- Theme4u—Lightning Console theme
- PortalDefault—Salesforce Customer Portal theme that applies to Customer Portals only and not to Experience Builder sites
- Webstore—AppExchange theme

**getUserEmail()**

Returns the current user’s email address.

**Signature**

```java
public static String getUserEmail()
```

**Return Value**

Type: **String**
Example

```java
String emailAddress = UserInfo.getUserEmail();
System.debug('Email address: ' + emailAddress);
```

**getUserId()**
Returns the context user's ID

**Signature**

```java
public static String getUserId()
```

**Return Value**
Type: String

**getUserName()**
Returns the context user's login name.

**Signature**

```java
public static String getUserName()
```

**Return Value**
Type: String

**getUserRoleId()**
Returns the context user's role ID.

**Signature**

```java
public static String getUserRoleId()
```

**Return Value**
Type: String

**getUserType()**
Returns the context user's type.

**Signature**

```java
public static String getUserType()
```
Return Value
Type: String

**hasPackageLicense(packageId)**
Returns `true` if the context user has a license to the managed package via a package license only. Otherwise, returns `false`.

**Signature**
```
public static Boolean hasPackageLicense(ID packageID)
```

**Parameters**
`packageID`  
Type: String

**Return Value**
Type: Boolean

**isCurrentUserLicensed(namespace)**
Returns `true` if the context user has a license to any managed package denoted by the namespace. Otherwise, returns `false`.

**Signature**
```
public static Boolean isCurrentUserLicensed(String namespace)
```

**Parameters**
`namespace`  
Type: String

**Return Value**
Type: Boolean

**Usage**
A `TypeException` is thrown if `namespace` is an invalid type.

**isCurrentUserLicensedForPackage(packageID)**
Returns `true` if the context user has a license to the managed package denoted by the package ID. Otherwise, returns `false`. If the context user has access, it’s determined either via the package license or a namespace permission set license for the package namespace.

**Signature**
```
public static Boolean isCurrentUserLicensedForPackage(ID packageID)
```
Parameters

`packageId`

Type: `String`

Return Value

Type: `Boolean`

Usage

Retrieve `packageId` at runtime, with the `getCurrentPackageId()` method. Then, use `packageId` to confirm that the contextual user is licensed to use that managed package.

A `TypeException` is thrown if `packageId` is an invalid type. A `SystemException` is thrown if `packageId` is the ID of an unlocked or unmanaged package, or if the contextual user doesn’t have a license to the managed package.

`isMultiCurrencyOrganization()`

Specifies whether the organization uses multiple currencies.

Signature

`public static Boolean isMultiCurrencyOrganization()`

Return Value

Type: `Boolean`

UserManagement Class

Contains methods to manage end users, for example, to register their verification methods, verify their identity, or remove their personal information.

Namespace

`System`

Usage

Let users register and deregister identity verification methods. Create custom Login and Verify pages for passwordless login and self-registration. Convert mobile phone numbers to the proper format before registering users. Scramble user data when users request that Salesforce remove their personal information.

This class is available in API version 43.0 and later.

IN THIS SECTION:

UserManagement Methods
UserManagement Methods

The following are methods for UserManagement.

IN THIS SECTION:

clone()

deregisterVerificationMethod(userId, method)
Deregisters an identity verification method. Use this method to let users delete an existing verification method.

formatPhoneNumber(countryCode, phoneNumber)
Formats a mobile phone number for a user. Call this method to ensure that the phone number is formatted properly before updating a user’s mobile phone number.

initPasswordlessLogin(userId, method)
Invokes a verification challenge for passwordless login when creating custom (Visualforce) Login and Verify pages for customers and partners.

initRegisterVerificationMethod(method)
Invokes a verification challenge for registering identity verification methods with a custom (Visualforce) page. Users can register either their email address or phone number.

initSelfRegistration(method, user)
Invokes a verification challenge for self-registration when creating a custom (Visualforce) Verify page for Experience Cloud self-registration.

initVerificationMethod(method)
Initiates a verification service for email, phone (SMS), and the Salesforce Authenticator verification methods.

initVerificationMethod(method, actionName, extras)
Initiates a verification service for email, phone (SMS), and the Salesforce Authenticator verification methods.

obfuscateUser(userId, username)
Scrambles users’ data on their request when they no longer want their personal data recognized in Salesforce. When you invoke the method for the user, the data becomes anonymous, and you can never recover it. Use this method to set the username to a specific value after it’s scrambled.

obfuscateUser(userId)
Scrambles users’ data on their request when they no longer want their personal data recognized in Salesforce. When you invoke the method for the user, the data becomes anonymous, and you can never recover it.

registerVerificationMethod(method, startUrl)
 Registers an identity verification method. Verification methods can be a time-based one-time password (TOTP), email or text verification code, Salesforce Authenticator, or U2F-compatible security key. End users register verification methods for themselves.

sendAsyncEmailConfirmation(userId, emailTemplateId, networkId, startUrl)
Send an email message to a user’s email address for verification. The message contains a verification link (URL) that the user clicks to verify the email address later on. You can send email verifications in bulk.

verifyPasswordlessLogin(userId, method, identifier, code, startUrl)
Completes a verification challenge during a passwordless login that uses a custom Verify page (Visualforce only). If the user who is trying to log in enters the verification code successfully, the user is logged in.
verifyRegisterVerificationMethod(code, method)
Completes registering a user’s email address or phone number as a verification method when customizing the identity verification process.

verifySelfRegistration(method, identifier, code, startUrl)
Completes a verification challenge when creating a custom (Visualforce) Verify page for Experience Cloud site self-registration. If the person who is attempting to register enters the verification code successfully, the user is created and logged in.

verifyVerificationMethod(identifier, code, method)
Completes the verification service for email, phone (SMS), Salesforce Authenticator, password, or time-based one-time password (TOTP) verification methods.

clonE()

Signature
custom Object clone()

Return Value
Type: User Management
deregisterVerificationMethod(userId, method)
Deregisters an identity verification method. Use this method to let users delete an existing verification method.

Signature
custom static void deregisterVerificationMethod(Id userId, Auth.VerificationMethod method)

Parameters
userId
Type: Id
User ID of the user deregistering the verification method.

method
Type: Auth.VerificationMethod
Verification method used to verify the identity of the user.

Return Value
Type: void
Usage

Use this method to deregister an existing identity verification method. For example, your users can deregister a phone number when their phone number changes. While only end users can register an identity verification method, you and your users can deregister one. Keep this behavior in mind when you implement a custom registration page.

This method is available in API version 43.0 and later.

ℹ️ Note: This method doesn’t support deregistering built-in authenticators.

`formatPhoneNumber(countryCode, phoneNumber)`

Formats a mobile phone number for a user. Call this method to ensure that the phone number is formatted properly before updating a user’s mobile phone number.

**Signature**

```
global static String formatPhoneNumber(String countryCode, String phoneNumber)
```

**Parameters**

- `countryCode`
  - Type: String
  - A valid country code.

- `phoneNumber`
  - Type: String
  - A mobile number that contains from 3 through 49 numeric characters, without the country code. For example, (415) 555-1234.

**Return Value**

- Type: String
  - Returns a user’s mobile phone number in the proper format.

Usage

Use this method to ensure a user’s mobile phone number is formatted as required by Salesforce. Then use the method’s return value to update the mobile field of the user’s record. This mobile number is used for SMS-based device activation. For example, mobile phone numbers are stored along with other identity verification methods in `Auth.VerificationMethod` enum. This method is introduced in API version 43.0. It isn’t available in earlier versions.

Here are some acceptable ways that users can enter their mobile number:

- +1, (415) 555-1234 (with plus signs, parentheses, and dashes)
- 1, 4155551234 (only numbers, no symbols)
- 1, 415-555-1234 (extra spaces)

Now, consider the following examples.

- Correct examples:
  - `formatPhoneNumber('1', '4155551234');`
  - `formatPhoneNumber('+1', '(415) 555-1234');`
- `formatPhoneNumber('1', '415-555-1234');`

  - Incorrect example, because the country code and mobile number aren’t separated:
    - `formatPhoneNumber(null, '+1 415-555-1234');`

  - Example that doesn’t generate an error, but likely won’t work as intended:
    - `formatPhoneNumber('+1', '+1 (415) 555-1234');`

### Format Phone Number Code Example

Here’s a code example that uses the `formatPhoneNumber` method. It gets the mobile number from the user and converts it to the format required by Salesforce. Then it updates the user’s record with the formatted mobile number.

```apex
global with sharing class PhoneRegistrationController {
    // Input variables
    global String countryCode {get; set;}
    global String phoneNumber {get; set;}

    global String addPhoneNumber() {
        if(countryCode == null) return 'Country code is required';
        if(phoneNumber == null) return 'Phone number is required';

        String userId = UserInfo.getUserId();
        User u = [SELECT Id FROM User WHERE Id=:userId LIMIT 1];
        String formatNum = System.UserManagement.formatPhoneNumber(countryCode, phoneNumber);

        u.MobilePhone = formatNum;
        update u;
        return null;
    }
}
```

As long as the country code and phone number are separated, `formatPhoneNumber` returns a value in the proper format.

### `initPasswordlessLogin(userId, method)`

Invokes a verification challenge for passwordless login when creating custom (Visualforce) Login and Verify pages for customers and partners.

#### Signature

`public static String initPasswordlessLogin(Id userId, Auth.VerificationMethod method)`

#### Parameters

- **userId**
  
  Type: `Id`
  
  ID of the user who’s logging in.

- **method**
  
  Type: `Auth.VerificationMethod`
Method used to verify the user’s identity, which can be EMAIL or SMS.

Return Value
Type: String
Identifier of the verification attempt.

Usage
Use this method along with its paired verifyPasswordlessLogin to customize the login experience with your own Visualforce Login and Verify pages. Invoke initPasswordlessLogin from the Login page where the user enters an email address or phone number.

Note: An alternative to using this combination of methods is to use Site.passwordlessLogin. Both approaches let you customize the Login page in Visualforce. With the paired methods, you can create custom Login and Verify pages. With Site.passwordlessLogin, Salesforce supplies the Verify page.

First call the initPasswordlessLogin method to initiate an authentication challenge. This method:
• Gets the user ID and verification method, such as EMAIL or SMS, from the Login page.
• Looks up the user and checks that the user is unique and active.
• Sends a verification code to the user.
• Adds an entry for the verification attempt to the Identity Verification History log, assigning an identifier to the verification attempt and setting the status to User challenged, waiting for response.
• Adds an entry for the Passwordless Login to the Login History log.
• Returns the identifier to verifyPasswordlessLogin to link the transactions.

Then call verifyPasswordlessLogin, which, if the user enters the verification code correctly, logs in the user.

Note: Users must verify their identity by email address or phone number before they can log in without a password. You can check whether the user is verified from the user’s detail page in Setup. Or you can check programmatically with TwoFactorMethodsInfo.

initRegisterVerificationMethod(method)
Invokes a verification challenge for registering identity verification methods with a custom (Visualforce) page. Users can register either their email address or phone number.

Signature
public static String initRegisterVerificationMethod(Auth.VerificationMethod method)

Parameters
method
Type: Auth.VerificationMethod
Method used to verify the user’s identity, which can be EMAIL or SMS.

Return Value
Type: String
The method returns an error message if the phone number is already registered, the user isn’t a customer or partner, or if the context isn’t an Experience Cloud site.

**Usage**

Use this method along with its paired `verifyRegisterVerificationMethod` on page 3576 to customize the process for registering a user’s verification method using a Visualforce Verify page.

First call the `initRegisterVerificationMethod` method to get the verification code sent to the user as input, and validate it. If the verification code isn’t valid, it returns an error message.

**Example**

Here’s a code example that registers a user’s phone number as a verification method. When the user enters a verification code on the Visualforce page, it invokes `registerUser()`. The method gets the User ID of the user who’s registering the verification method and the user’s phone number. It also gets the user’s registration status to check whether the phone number is verified already. If the user is registered with a different phone number, the number is updated.

```apex
public void registerUser() {
    try {
        exceptionText = ''; 
        String userId = UserInfo.getUserId();
        User u = [Select MobilePhone, Id from User Where Id=:userId];
        currPhone = u.MobilePhone;
        mobilePhone = getFormattedSms(mobilePhone);
        if (mobilePhone != null && mobilePhone != '') {
            u.MobilePhone = mobilePhone;
            update u;
            // We're updating the email and phone number before verifying. Roll back
            // the change in the verify API if it is unsuccessful.
            exceptionText = System.
            UserManagement.initRegisterVerificationMethod(Auth.VerificationMethod.SMS);
            if(exceptionText!= null && exceptionText!='') {
                isInit = false;
                showInitException = true;
            } else {
                isInit = false;
                isVerify = true;
            }
        } else {
            showInitException = true;
        }
    } catch (Exception e) {
        exceptionText = e.getMessage();
        isInit = false;
        showInitException = true;
    }
}

public void verifyUser() {
    // Take the user’s input for the code sent to their phone number
    exceptionText = System.
    UserManagement.
    verifyRegisterVerificationMethod(code, Auth.VerificationMethod.SMS);
    if(exceptionText != null && exceptionText!=''){
        isVerify = false;
        showInitException = true;
    }
}
```
showInitException = true;
} else {
    //Success
}

initSelfRegistration(method, user)
Invokes a verification challenge for self-registration when creating a custom (Visualforce) Verify page for Experience Cloud self-registration.

Signature
public static String initSelfRegistration(Auth.VerificationMethod method, User user)

Parameters
method
Type: Auth.VerificationMethod
Method used to verify the identity of the user, which can be EMAIL or SMS.

user
Type: User
User object to insert after successful registration.

Return Value
Type: String
Identifier of the registration attempt.

Usage
By default, when users sign up for your Experience Cloud site with an email address or phone number, Salesforce sends them a verification code. At the same time, it generates a Verify page for users to confirm their identity. You can replace the default Salesforce Verify page with your own Visualforce page and then invoke the verification process.

Call this method to initiate the authentication challenge, and include a User object to insert if the registration is successful. The method returns the identifier for the self-registration attempt.

Note: If you specify a language in the LanguageLocaleKey field on the User object, Salesforce uses this language for verification email and SMS messages.

Then call verifySelfRegistration, which, if the user enters the verification code correctly, logs in the user.

Example
This code contains the result of a verification challenge that registers a new user.

```java
String id = System.UserManagement.initSelfRegistration
(Auth.VerificationMethod.SMS, user);
Auth.VerificationResult res = System.UserManagement.verifySelfRegistration
(Auth.VerificationMethod.SMS, id, '123456', null);
if(res.success == true){
```
initVerificationMethod (method)

Initiates a verification service for email, phone (SMS), and the Salesforce Authenticator verification methods.

**Signature**

```java
public static String initVerificationMethod(Auth.VerificationMethod method)
```

**Parameters**

- `method`
  - Type: `Auth.VerificationMethod`
  - Method used to initiate a verification service for EMAIL, SMS, or SALESFORCE_AUTHENTICATOR verification methods.

**Return Value**

- Type: `String`
  - The returned identifier must be passed into `verifyVerificationMethod`.

**Usage**

Use this method along with its paired `verifyVerificationMethod` to customize a verification service for EMAIL, SMS, or SALESFORCE_AUTHENTICATOR verification methods. The returned identifier from `initVerificationMethod` must be passed into `verifyVerificationMethod`.

First invoke the `initVerificationMethod` method to send a verification code to the user’s email or phone number, or to send a push notification to the Salesforce Authenticator. The user then enters the code or approves the push notification. If the verification code isn’t valid or the push notification isn’t approved, the service returns an error message.

**Email Example**

This example shows multi-factor authentication using email.

```java
public void initVerification() {
    // user will receive code on their registered verified email
    identifier = UserManagement.initVerificationMethod(Auth.VerificationMethod.EMAIL);
}

public Auth.VerificationResult verifyVerification() {
    // requiring identifier from the initVerification
    // the code will need to be entered in this method
    return UserManagement.verifyVerificationMethod(identifier, code ,
        Auth.VerificationMethod.EMAIL);
}
```

initVerificationMethod (method, actionName, extras)

Initiates a verification service for email, phone (SMS), and the Salesforce Authenticator verification methods.
Signature

public static String initVerificationMethod(Auth.VerificationMethod method, String actionName, Map<String,String> extras)

Parameters

method
Type: Auth.VerificationMethod
Method used to initiate a verification service for EMAIL, SMS, or SALESFORCE_AUTHENTICATOR verification methods.

actionName
Type: String
For the SALESFORCE_AUTHENTICATOR verification method only, the name of the action to display on the Salesforce Authenticator, such as Connect to My Salesforce Org. The default action name is Apex-Defined Activity.

extras
Type: Map<String,String>
For the SALESFORCE_AUTHENTICATOR verification method only, the following extra settings.

- secure_device_required--If set to true, the user’s device must be secured. For example, the user must enter the device’s passcode to approve the request. Default setting is false.
- challenge_required--If set to true, the user must complete a biometric challenge, such as face recognition, on the device to approve the request. Default setting is false.

Return Value

Type: String
The returned identifier must be passed into verifyVerificationMethod method.

Usage

Use this method along with its paired verifyVerificationMethod to customize a verification service for EMAIL, SMS, or SALESFORCE_AUTHENTICATOR verification methods. The returned identifier from initVerificationMethod must be passed into verifyVerificationMethod method.

First invoke the initVerificationMethod method to send a verification code to the user’s email or phone number, or to send a push notification to the Salesforce Authenticator. The user then enters the code or approves the push notification. If the verification code isn’t valid or the push notification isn’t approved, the service returns an error message.

Salesforce Authenticator Example

This example shows multi-factor authentication (MFA) using the Salesforce Authenticator mobile app. In this example, the actionName parameter is set to the default setting and the extra parameter settings are set to false.

```java
public void initVerification() {
    // user will receive push notification on their registered MFA devices
    identifier = UserManagement.initVerificationMethod(Auth.VerificationMethod.SALESFORCE_AUTHENTICATOR);
}

public Auth.VerificationResult verifyVerification() {
```
This example shows multi-factor authentication using Salesforce Authenticator. In this example, the `actionName` parameter is set to `Connect to My Salesforce Org` and the `challenge_required` extra parameter is set to `true`.

```apex
public void initVerification() {
    Map<String, String> extras = new Map<String, String>();
    extras.put('challenge_required', 'true');
    // user will receive push notification in their registered MFA devices
    identifier = UserManagement.initVerificationMethod(Auth.VerificationMethod.SALESFORCE_AUTHENTICATOR, 'Connect to My Salesforce Org', extras);
}

public Auth.VerificationResult verifyVerification() {
    // requiring identifier from the initVerification
    // user will need to take the action on their registered MFA devices
    return UserManagement.verifyVerificationMethod(identifier, '', Auth.VerificationMethod.SALESFORCE_AUTHENTICATOR);
}
```

**obfuscateUser(userId, username)**

Scrambles users’ data on their request when they no longer want their personal data recognized in Salesforce. When you invoke the method for the user, the data becomes anonymous, and you can never recover it. Use this method to set the username to a specific value after it’s scrambled.

**Signature**

```apex
public static void obfuscateUser(Id userId, String username)
```

**Parameters**

- **userId**
  - Type: Id
  - ID of the user whose data this method scrambles.

- **username**
  - Type: String
  - The username after the user’s data is scrambled. Sets the value of the scrambled username to a specific string.

**Return Value**

Type: void

**Usage**

This method is introduced in API version 43.0. It isn't available in earlier versions.
You can use the `obfuscateUser` method to protect the personal information of your org's users. When invoked, Salesforce permanently scrambles the user’s object data and replaces it with random character strings. The user’s detail page exists, but the fields contain meaningless strings of characters. Salesforce merely obfuscates (scrambles) personal data because you can’t delete a user in Salesforce; you can only disable or deactivate a user. In other words, the user record remains in the database and this method performs a soft delete.

**Note:** Take care when using this method. The users’ data becomes anonymous and can never be recovered.

### Considerations
- This method requires that the org’s User Management setting, *Scramble Specific Users' Data*, is enabled from Setup.
- This method affects the standard fields of the user object—excluding a few fields such as the user ID, timezone, locale, and profile.
- It is recommended that you note the user’s ID and other attributes for post processing, such as the email address, if you want to send the user a confirmation.
- This method changes only the user object. The association between the user and other objects is removed, but no other objects are changed. For example, contact, ThirdPartyAccountLink (TPAL), and user password authentication (UPA) objects remain unchanged.

**Note:** Assure your admins that invoking this method doesn’t trigger an email change notification.

This method is part of our effort to protect users’ personal data and privacy. For more information on what you can do to actively protect user data, see Data Protection and Privacy in Salesforce Help.

**obfuscateUser** *(userId)*

Scrambles users’ data on their request when they no longer want their personal data recognized in Salesforce. When you invoke the method for the user, the data becomes anonymous, and you can never recover it.

### Signature

```java
public static void obfuscateUser(Id userId)
```

### Parameters

- **userId**
  - Type: `Id`
  - ID of the user whose data this method scrambles.

### Return Value

- Type: `void`

### Usage

This method is introduced in API version 43.0. It isn’t available in earlier versions.

You can use the `obfuscateUser` method to protect the personal information of your org’s users. When invoked, Salesforce permanently scrambles the user’s object data and replaces it with random character strings. The user’s detail page exists, but the fields contain meaningless strings of characters. Salesforce merely obfuscates (scrambles) personal data because you can’t delete a user in Salesforce; you can only disable or deactivate a user. In other words, the user record remains in the database and this method performs a soft delete.

**Note:** Take care when using this method. The users’ data becomes anonymous and can never be recovered.
Considerations

- This method requires that the org’s User Management setting, Scramble Specific Users’ Data, is enabled from Setup.
- This method affects the standard fields of the user object—excluding a few fields such as the user ID, timezone, locale, and profile.
- If you want to send the user a confirmation, it’s recommended that you note the user’s ID and other attributes for post processing, such as the email address.
- This method changes only the user object. The association between the user and other objects is removed, but no other objects are changed. For example, contact, ThirdPartyAccountLink (TPAL), and user password authentication (UPA) objects remain unchanged.

Note: Assure your admins that invoking this method doesn’t trigger an email change notification.

This method is part of our effort to protect users’ personal data and privacy. For more information on what you can do to actively protect user data, see Data Protection and Privacy in Salesforce Help.

ObfuscateUser Code Example

```java
public class UserManagementController{
    public List<User> users {get; set;}

    public UserManagementController()
    {
        Profile p = [select id from profile where name = 'Customer Community User'];
        users = [select username, id from User where profileId=:p.id AND isactive=true];
    }

    //Use method with extreme caution. Data can’t be recovered.
    @InvocableMethod(label='User Management' description='Obfuscate User data and more')
    static public void obfuscate(List<User> users)
    {
        String uid = ApexPages.currentPage().getParameters().get('uid');
        if(uid == null)
            return;
        User u = [select contactId from user where id=:uid];
        System.UserManagement.obfuscateUser(uid);
    }
}
```

**registerVerificationMethod(method, startUrl)**

Registers an identity verification method. Verification methods can be a time-based one-time password (TOTP), email or text verification code, Salesforce Authenticator, or U2F-compatible security key. End users register verification methods for themselves.

**Signature**

```java
```
Parameters

method
  Type: Auth.VerificationMethod
  Verification method used to verify the identity of the user.

startUrl
  Type: String
  Path to the page that users see after they log in.

Return Value

Type: System.PageReference

Usage

Use this method to enable users to complete identity verification, such as multi-factor authentication (MFA), or to log in to their Experience Cloud site without a password. Users register these methods to verify their identity when logging in. You create a custom registration page when implementing mobile-centric passwordless logins. See VerifyPasswordlessLogin.

The PageReference returned by registerVerificationMethod redirects the user to the Salesforce Verify page. If the user enters the correct code, the user is redirected to the Experience Cloud site page specified by the start URL. For example:

```java
PageReference pr = System.UserManagement.registerVerificationMethod(Auth.VerificationMethod.TOTP, startUrl);
PageReference p = System.UserManagement.deregisterVerificationMethod(userId, Auth.VerificationMethod.SALESFORCE_AUTHENTICATOR);
```

This method is available in API version 43.0 and later.

Note: As a security measure, when users add or update mobile numbers in their detail page, they must log in again to verify their identity. As a result, unsaved changes in the app are lost. To disable this security measure, contact Salesforce Support.

sendAsyncEmailConfirmation(userId, emailTemplateId, networkId, startUrl)

Send an email message to a user's email address for verification. The message contains a verification link (URL) that the user clicks to verify the email address later on. You can send email verifications in bulk.

Signature

public static Boolean sendAsyncEmailConfirmation(String userId, String emailTemplateId, String networkId, String startUrl)

Parameters

userId
  Type: String
  ID of the user to receive the email confirmation.
emailTemplateId
Type: String
ID of the email template in which the verification link is defined.

networkId
Type: String
ID of the Experience Cloud site.

startUrl
Type: String
The user is redirected to this page after verification, with a success or error message as the parameter. If null, the user is redirected to the login page.

Return Value
Type: Boolean
Indicates whether sending the email message succeeded or failed.

Usage
Sending an async email message is good practice to ensure that users are registered with a valid email address that they truly own. To determine which users receive an email with the verification link, check whether the User Verified Email field in the User detail page is set to true. You can also get this information from the TwoFactorMethodsInfo API.
Send async email verification to customers and partners to verify their email address. These users must verify their email address before they can log in with email OTP (passwordless login).
The error code and description are passed as query parameters so that you can process any errors when building a custom landing page.

Example
```
System.UserManagement.sendAsyncEmailConfirmation('005RM000001a0Ox',
'00XRM000000hxnG','0DBRM000000015i', '/s/contactsupport');
```

verifyPasswordlessLogin(userId, method, identifier, code, startUrl)
Completes a verification challenge during a passwordless login that uses a custom Verify page (Visualforce only). If the user who is trying to log in enters the verification code successfully, the user is logged in.

Signature
```
public static Auth.VerificationResult verifyPasswordlessLogin(Id userId,
Auth.VerificationMethod method, String identifier, String code, String startUrl)
```

Parameters
userId
Type: Id
ID of the user who’s logging in.
method
Type: Auth.VerificationMethod
Method used to verify the identity of the user, which can be either EMAIL or SMS.

identifier
Type: String
ID of the verification attempt received from the initPasswordlessLogin method.

code
Type: String
Code used to verify the identity of the user.

startUrl
Type: String
The page where the user is directed after successful login.

Return Value
Type: Auth.VerificationResult
Result of the verification challenge, which includes the message displayed, and where the user is directed if they enter the verification code correctly.

Usage
Call this method to complete the passwordless login authentication process. It validates the verification method and verification code. It also checks that the identifier is the same as the one returned by initPasswordlessLogin on page 3565.

Example
For an example, see Auth.VerificationResult.

`verifyRegisterVerificationMethod(code, method)`
Completes registering a user’s email address or phone number as a verification method when customizing the identity verification process.

Signature
`public static String verifyRegisterVerificationMethod(String code, Auth.VerificationMethod method)`

Parameters
`code`
Type: String
Code used to verify the identity of the user.

`method`
Type: Auth.VerificationMethod
Method used to verify the identity of the user, which can be either EMAIL or SMS.
Return Value

Type: String

If the user enters an incorrect verification code, the method returns an error message.

Usage

Call verifyRegisterVerificationMethod to complete the process of registering a user's verification method. This method checks whether the user entered the correct verification code. If the verification code is correct, the method

• Confirms that the user entered the correct verification code
• From the user's detail page, updates the user's verification method status (sets the verification bit)
• Sends an email to the user confirming that a verification method has been added to their record

If the verification code is incorrect, an error message is returned.

Note: If users want to change their email address after registering one, don’t use the initRegisterVerificationMethod and verify RegisterVerificationMethod methods. To enable automatic identity verification for email address changes, from the Identity Verification Setup page, select the field Require email confirmations for email address changes (applies to users in Experience Builder sites).

Example

Here’s a code example that registers a user's phone number as a verification method. When the user enters a verification code on the Visualforce page, it invokes registerUser(). The method gets the User ID of the user who’s registering the verification method and the user’s phone number. It also gets the user’s registration status to check whether the phone number is verified already. If the user is registered with a different phone number, the number is updated.

```apex
public void registerUser() {
    try {
        exceptionText = ''; 
        String userId = UserInfo.getUserId();
        User u = [Select MobilePhone, Id from User Where Id=:userId];
        currPhone = u.MobilePhone;
        mobilePhone = getFormattedSms(mobilePhone);
        if (mobilePhone != null && mobilePhone != '') {
            u.MobilePhone = mobilePhone;
            update u;
            // We're updating the email and phone number before verifying. Roll back // the change in the verify API if it is unsuccessful.
            exceptionText = System.
            UserManagement.initRegisterVerificationMethod(Auth.VerificationMethod.SMS);
            if(exceptionText!= null && exceptionText!=''){
                isInit = false;
                showInitException = true;
            } else {
                isInit = false;
                isVerify = true;
            }
        } else {
            showInitException = true;
        }
    } catch (Exception e) {
        exceptionText = e.getMessage();
    }
} 
```

3577
isInit = false;
showInitException = true;
}
}

public void verifyUser() {
    // Take the user’s input for the code sent to their phone number
    exceptionText = System.UserManagement.
    verifyRegisterVerificationMethod(code, Auth.VerificationMethod.SMS);
    if(exceptionText != null && exceptionText !=''){showInitException = true;
    } else {
        //Success
    }
}

**verifySelfRegistration(method, identifier, code, startUrl)**

Completes a verification challenge when creating a custom (Visualforce) Verify page for Experience Cloud site self-registration. If the person who is attempting to register enters the verification code successfully, the user is created and logged in.

**Signature**

public static Auth.VerificationResult verifySelfRegistration(Auth.VerificationMethod method, String identifier, String code, String startUrl)

**Parameters**

**method**

Type: Auth.VerificationMethod

Method used to verify the identity of the user, which can be either EMAIL or SMS.

**identifier**

Type: String

The unique identifier received from the initSelfRegistration method.

**code**

Type: String

Code used to verify the identity of the user.

**startUrl**

Type: String

The page where the user is directed after successful self-registration.

**Return Value**

Type: Auth.VerificationResult

Result of the verification challenge, which includes the message displayed, and where the user is directed when they enter the verification code correctly.
Usage

By default, when users sign up for your Experience Cloud site with an email address or phone number, Salesforce sends them a verification code and generates a Verify page. This Verify page is where users enter the verification code to confirm their identity. You can replace this Salesforce-generated Verify page with a custom Verify page that you create with Visualforce. Then you invoke the verification process with Apex methods.

First, call the `initSelfRegistration` method, which returns the identifier of the user to create. Then call this `verifySelfRegistration` method to complete the verification process. If the user enters the verification code correctly, the user is created and directed to the page specified in the `startURL`.

This method returns the verification result, which contains the verification status and, if the user is created, the session ID. If the verification method is SMS, the User object must contain a properly formatted mobile number, which is country code, space, and then phone number, for example, +1 1234567890. Use `System.UserManagement.formatPhoneNumber` to ensure that the phone number is formatted correctly.

Example

This code contains the result of a verification challenge that registers a new user.

```java
String id = System.UserManagement.initSelfRegistration
(Auth.VerificationMethod.SMS, user);
Auth.VerificationResult res = System.UserManagement.verifySelfRegistration
(Auth.VerificationMethod.SMS, id, '123456', null);
if(res.success == true){
    //redirect
}
```

`verifyVerificationMethod(identifier, code, method)`

Completes the verification service for email, phone (SMS), Salesforce Authenticator, password, or time-based one-time password (TOTP) verification methods.

Signature

```java
public static VerificationResult verifyVerificationMethod(String identifier, String code, Auth.VerificationMethod method)
```

Parameters

- **identifier**
  - Type: `String`
  - Identifier returned from `initVerificationMethod` for EMAIL, SMS, and SALESFORCE_AUTHENTICATOR.

- **code**
  - Type: `String`
  - Code used to verify the user's identity for EMAIL, SMS, or PASSWORD.

- **method**
  - Type: `Auth.VerificationMethod`
  - Method used to verify the user's identity, which can be EMAIL, PASSWORD, SALESFORCE_AUTHENTICATOR, SMS, or TOTP.
Return Value
Type: VerificationResult

Usage
Use this method along with its paired initVerificationMethod to customize a verification service for EMAIL, SMS, or SALESFORCE_AUTHENTICATOR verification methods. Or use this method alone to provide a complete verification service for PASSWORD and TOTP verification methods.

This method checks whether the user entered the correct verification code or password. If the verification code or password is correct, the method verifies the user’s identity.

If the verification code or password isn’t valid, the service returns an error message.

Examples
This example shows multi-factor authentication using email.

```java
public void initVerification() {
    // user will receive code on their registered verified email
    identifier = UserManagement.initVerificationMethod(Auth.VerificationMethod.EMAIL);
}

public Auth.VerificationResult verifyVerification() {
    // requiring identifier from the initVerification
    // the code will need to be entered in this method
    return UserManagement.verifyVerificationMethod(identifier, code ,
        Auth.VerificationMethod.EMAIL);
}
```

The next two examples show multi-factor authentication using only the verifyVerificationMethod for password and TOTP verifications.

```java
public Auth.VerificationResult verifyVerification() {
    // user will enter their password as a param in the verifyVerificationMethod for password verification method
    return UserManagement.verifyVerificationMethod('', password ,
        Auth.VerificationMethod.PASSWORD);
}

public Auth.VerificationResult verifyVerification() {
    // user will enter their registered time-based one-time password (TOTP) code (token)
    return UserManagement.verifyVerificationMethod('', code ,
        Auth.VerificationMethod.TOTP);
}
```

Version Class
Use the Version methods to get the version of a first-generation managed package, and to compare package versions.

Namespace
System
Usage

A package version is a number that identifies the set of components uploaded in a package. The version number has the format `majorNumber.minorNumber.patchNumber` (for example, 2.1.3). The major and minor numbers increase to a chosen value during every major release. The `patchNumber` is generated and updated only for a patch release.

A called component can check the version against which the caller was compiled using the `System.requestVersion` method and behave differently depending on the caller’s expectations. This allows you to continue to support existing behavior in classes and triggers in previous package versions while continuing to evolve the code.

The value returned by the `System.requestVersion` method is an instance of this class with a two-part version number containing a major and a minor number. Since the `System.requestVersion` method doesn’t return a patch number, the patch number in the returned `Version` object is null.

The `System.Version` class can also hold also a three-part version number that includes a patch number.

Example

This example shows how to use the methods in this class, along with the `requestVersion` method, to determine the managed package version of the code that is calling your package.

```java
if (System.requestVersion() == new Version(1,0))
{
    // Do something
}
if ((System.requestVersion().major() == 1)
    && (System.requestVersion().minor() > 0)
    && (System.requestVersion().minor() <=9))
{
    // Do something different for versions 1.1 to 1.9
}
else if (System.requestVersion().compareTo(new Version(2,0)) >= 0)
{
    // Do something completely different for versions 2.0 or greater
}
```

IN THIS SECTION:

- Version Constructors
- Version Methods

Version Constructors

The following are constructors for `Version`.

IN THIS SECTION:

- `Version(major, minor)`
  Creates a new instance of the `Version` class as a two-part package version using the specified major and minor version numbers.
- `Version(major, minor, patch)`
  Creates a new instance of the `Version` class as a three-part package version using the specified major, minor, and patch version numbers.
**Version(major, minor)**

Creates a new instance of the `Version` class as a two-part package version using the specified major and minor version numbers.

**Signature**

```java
public Version(Integer major, Integer minor)
```

**Parameters**

- **major**
  - Type: `Integer`
  - The major version number.

- **minor**
  - Type: `Integer`
  - The minor version number.

**Version(major, minor, patch)**

Creates a new instance of the `Version` class as a three-part package version using the specified major, minor, and patch version numbers.

**Signature**

```java
public Version(Integer major, Integer minor, Integer patch)
```

**Parameters**

- **major**
  - Type: `Integer`
  - The major version number.

- **minor**
  - Type: `Integer`
  - The minor version number.

- **patch**
  - Type: `Integer`
  - The patch version number.

**Version Methods**

The following are methods for `Version`. All are instance methods.

**IN THIS SECTION:**

- `compareTo(version)`
  - Compares the current version with the specified version.
major()
Returns the major package version of the calling code.

minor()
Returns the minor package version of the calling code.

patch()
Returns the patch package version of the calling code or `null` if there is no patch version.

**compareTo(version)**
Compares the current version with the specified version.

**Signature**
```
public Integer compareTo(System.Version version)
```

**Parameters**

- `version`
  Type: `System.Version`

**Return Value**
Type: `Integer`
Returns one of the following values:
- zero if the current package version is equal to the specified package version
- an Integer value greater than zero if the current package version is greater than the specified package version
- an Integer value less than zero if the current package version is less than the specified package version

**Usage**
If a two-part version is being compared to a three-part version, the patch number is ignored and the comparison is based only on the major and minor numbers.

**major()**
Returns the major package version of the calling code.

**Signature**
```
public Integer major()
```

**Return Value**
Type: `Integer`

**minor()**
Returns the minor package version of the calling code.
Signature
public Integer minor()

Return Value
Type: Integer

patch()
Returns the patch package version of the calling code or null if there is no patch version.

Signature
public Integer patch()

Return Value
Type: Integer

WebServiceCallout Class
Enables making callouts to SOAP operations on an external Web service. This class is used in the Apex stub class that is auto-generated from a WSDL.

Namespace
System

IN THIS SECTION:
WebServiceCallout Methods

SEE ALSO:
Apex Developer Guide: SOAP Services: Defining a Class from a WSDL Document

WebServiceCallout Methods
The following is the static method for WebServiceCallout.

IN THIS SECTION:
invoke(stub, request, response, infoArray)
Invokes an external SOAP web service operation based on an Apex class that is auto-generated from a WSDL.

invoke(stub, request, response, infoArray)
Invokes an external SOAP web service operation based on an Apex class that is auto-generated from a WSDL.
Signature

```java
public static void invoke(Object stub, Object request, Map<String, Object> response,
List<String> infoArray)
```

Parameters

**stub**

Type: Object

An instance of the Apex class that is auto-generated from a WSDL (the stub class).

**request**

Type: Object

The request to the external service. The request is an instance of a type that is created as part of the auto-generated stub class.

**response**

Type: `Map<String, Object>`

A map of key-value pairs that represent the response that the external service sends after receiving the request. In each pair, the key is a response identifier. The value is the response object, which is an instance of a type that is created as part of the auto-generated stub class.

**infoArray**

Type: `String[]`

An array of strings that contains information about the callout—web service endpoint, SOAP action, request, and response. The order of the elements in the array matters.

- Element at index 0 ([0]): One of the following options for identifying the URL of the external web service.
  - Endpoint URL. For example: 'http://YourServer/YourService'
  - Named credential URL, which contains the scheme `callout`, the name of the named credential, and optionally, an appended path. For example: 'callout:MyNamedCredential/some/path'
- Element at index 1 ([1]): The SOAP action. For example: 'urn:dotnet.callouttest.soap.sforce.com/EchoString'
- Element at index 2 ([2]): The request namespace. For example: 'http://doc.sample.com/docSample'
- Element at index 3 ([3]): The request name. For example: 'EchoString'
- Element at index 4 ([4]): The response namespace. For example: 'http://doc.sample.com/docSample'
- Element at index 5 ([5]): The response name. For example: 'EchoStringResponse'
- Element at index 6 ([6]): The response type. For example: 'docSample.EchoStringResponse_element'

Return Value

Type: Void

SEE ALSO:

*Apex Developer Guide: Named Credentials as Callout Endpoints*

**WebServiceMock Interface**

Enables sending fake responses when testing Web service callouts of a class auto-generated from a WSDL.
Namespace

System

Usage

For an implementation example, see Test Web Service Callouts.

WebServiceMock Methods

The following are methods for WebServiceMock.

IN THIS SECTION:

    doInvoke(stub, soapRequest, responseMap, endpoint, soapAction, requestName, responseNamespace, responseName, responseType)

    The implementation of this method is called by the Apex runtime to send a fake response when a Web service callout is made after Test.setMock has been called.

    doInvoke(stub, soapRequest, responseMap, endpoint, soapAction, requestName, responseNamespace, responseName, responseType)

    The implementation of this method is called by the Apex runtime to send a fake response when a Web service callout is made after Test.setMock has been called.

Signature

public Void doInvoke(Object stub, Object soapRequest, Map<String, Object> responseMap, String endpoint, String soapAction, String requestName, String responseNamespace, String responseName, String responseType)

Parameters

stub

    Type: Object
    An instance of the auto-generated class.

soapRequest

    Type: Object
    The SOAP Web service request being invoked.

responseMap

    Type: Map<String, Object>
    A collection of key/value pairs representing the response to send for the request.
    When implementing this interface, set the responseMap argument to a key/value pair representing the response desired.

endpoint

    Type: String
    The endpoint URL for the request.
soapAction
   Type: String
   The requested SOAP operation.

requestName
   Type: String
   The requested SOAP operation name.

responseNamespace
   Type: String
   The response namespace.

responseName
   Type: String
   The name of the response element as defined in the WSDL.

responseType
   Type: String
   The class for the response as defined in the auto-generated class.

Return Value
   Type: Void

Usage

**XmlStreamReader Class**

The XmlStreamReader class provides methods for forward, read-only access to XML data. You can pull data from XML or skip unwanted events. You can parse nested XML content that’s up to 50 nodes deep.

Namespace

System

Usage

The XmlStreamReader class is similar to the XMLStreamReader utility class from StAX (Streaming API for XML). StAX is an API to read and write XML documents, originating from the Java programming language community.

**Note:** The XmlStreamReader class in Apex is based on its counterpart in Java. See [Java XMLStreamReader class](#).

IN THIS SECTION:

XmlStreamReader Constructors
XmlStreamReader Methods

SEE ALSO:
Apex Developer Guide: Reading XML Using Streams

XmlStreamReader Constructors

The following are constructors for XmlStreamReader.

IN THIS SECTION:
XmlStreamReader(xmlInput)
Creates a new instance of the XmlStreamReader class for the specified XML input.

XmlStreamReader(xmlInput)
Creates a new instance of the XmlStreamReader class for the specified XML input.

Signature
public XmlStreamReader(String xmlInput)

Parameters
xmlInput:
Type: String
The XML string input.

XmlStreamReader Methods

The following are methods for XmlStreamReader. All are instance methods.

IN THIS SECTION:
getAttributeCount()
Returns the number of attributes on the start element, excluding namespace definitions.
getAttributeLocalName(index)
Returns the local name of the attribute at the specified index.
getAttributeNamespace(index)
Returns the namespace URI of the attribute at the specified index.
getAttributePrefix(index)
Returns the prefix of this attribute at the specified index.
getAttributeType(index)
Returns the XML type of the attribute at the specified index.
getAttributeValue(namespaceUri, localName)
Returns the value of the attribute in the specified localName at the specified URI.
getAttributeValueAt(index)
Returns the value of the attribute at the specified index.

getEventType()
Returns the type of XML event the cursor is pointing to.

getLocalName()
Returns the local name of the current event.

getLocation()
Return the current location of the cursor.

getNamespace()
If the current event is a start element or end element, this method returns the URI of the prefix or the default namespace.

getNamespaceCount()
Returns the number of namespaces declared on a start element or end element.

getNamespacePrefix(index)
Returns the prefix for the namespace declared at the index.

getNamespaceURI(prefix)
Return the URI for the given prefix.

getNamespaceURIAt(index)
Returns the URI for the namespace declared at the index.

getPIData()
Returns the data section of a processing instruction.

getPITarget()
Returns the target section of a processing instruction.

getPrefix()
Returns the prefix of the current XML event or null if the event does not have a prefix.

getText()
Returns the current value of the XML event as a string.

getVersion()
Returns the XML version specified on the XML declaration. Returns null if none was declared.

hasName()
Returns true if the current XML event has a name. Returns false otherwise.

hasNext()
Returns true if there are more XML events and false if there are no more XML events.

hasText()
Returns true if the current event has text, false otherwise.

isCharacters()
Returns true if the cursor points to a character data XML event. Otherwise, returns false.

isEndElement()
Returns true if the cursor points to an end tag. Otherwise, it returns false.

isStartElement()
Returns true if the cursor points to a start tag. Otherwise, it returns false.
isWhiteSpace()  
Returns true if the cursor points to a character data XML event that consists of all white space. Otherwise it returns false.

next()  
Reads the next XML event. A processor may return all contiguous character data in a single chunk, or it may split it into several chunks. Returns an integer which indicates the type of event.

nextTag()  
Skips any white space (the isWhiteSpace method returns true), comment, or processing instruction XML events, until a start element or end element is reached. Returns the index for that XML event.

setCoalescing(returnAsSingleBlock)  
If you specify true for returnAsSingleBlock, text is returned in a single block, from a start element to the first end element or the next start element, whichever comes first. If you specify it as false, the parser may return text in multiple blocks.

setNamespaceAware(isNamespaceAware)  
If you specify true for isNamespaceAware, the parser recognizes namespace. If you specify it as false, the parser does not. The default value is true.

toString()  
Returns a string containing the length of the input XML given to XmlStreamReader and the first 50 characters of the input XML.

getAttributeCount()  
Returns the number of attributes on the start element, excluding namespace definitions.

**Signature**

public Integer getAttributeCount()  

**Return Value**

Type: Integer

**Usage**

This method is only valid on a start element or attribute XML events. The count for the number of attributes for an attribute XML event starts with zero.

getAttributeLocalName(index)  
Returns the local name of the attribute at the specified index.

**Signature**

public String getAttributeLocalName(Integer index)

**Parameters**

index  
Type: Integer
Return Value
Type: String

Usage
If there is no name, an empty string is returned. This method is only valid with start element or attribute XML events.

getAttributeNamespace(index)
Returns the namespace URI of the attribute at the specified index.

Signature
public String getAttributeNamespace(Integer index)

Parameters
index
Type: Integer

Return Value
Type: String

Usage
If no namespace is specified, null is returned. This method is only valid with start element or attribute XML events.

getAttributePrefix(index)
Returns the prefix of this attribute at the specified index.

Signature
public String getAttributePrefix(Integer index)

Parameters
index
Type: Integer

Return Value
Type: String

Usage
If no prefix is specified, null is returned. This method is only valid with start element or attribute XML events.
**getAttributeType(index)**
Returns the XML type of the attribute at the specified index.

**Signature**
```
public String getAttributeType(Integer index)
```

**Parameters**
- **index**
  Type: Integer

**Return Value**
Type: String

**Usage**
For example, `id` is an attribute type. This method is only valid with start element or attribute XML events.

**getAttributeValue(namespaceUri, localName)**
Returns the value of the attribute in the specified `localName` at the specified URI.

**Signature**
```
public String getAttributeValue(String namespaceUri, String localName)
```

**Parameters**
- **namespaceUri**
  Type: String
- **localName**
  Type: String

**Return Value**
Type: String

**Usage**
Returns `null` if the value is not found. You must specify a value for `localName`. This method is only valid with start element or attribute XML events.

**getAttributeValueAt(index)**
Returns the value of the attribute at the specified index.
Signature

```java
public String getAttributeValueAt(Integer index)
```

Parameters

```java
index
Type: Integer
```

Return Value

Type: String

Usage

This method is only valid with start element or attribute XML events.

`getEventType()`

Returns the type of XML event the cursor is pointing to.

Signature

```java
public System.XmlTag getEventType()
```

Return Value

Type: System.XmlTag

**XmlTag Enum**

The values for XmlTag are:

- ATTRIBUTE
- CDATA
- CHARACTERS
- COMMENT
- DTD
- END_DOCUMENT
- END_ELEMENT
- ENTITY_DECLARATION
- ENTITY_REFERENCE
- NAMESPACE
- NOTATION_DECLARATION
- PROCESSING_INSTRUCTION
- SPACE
- START_DOCUMENT
- START_ELEMENT
**getLocalName()**

Returns the local name of the current event.

**Signature**

```java
public String getLocalName()
```

**Return Value**

Type: String

**Usage**

For start element or end element XML events, it returns the local name of the current element. For the entity reference XML event, it returns the entity name. The current XML event must be start element, end element, or entity reference.

**getLocation()**

Return the current location of the cursor.

**Signature**

```java
public String getLocation()
```

**Return Value**

Type: String

**Usage**

If the location is unknown, returns -1. The location information is only valid until the next method is called.

**getNamespace()**

If the current event is a start element or end element, this method returns the URI of the prefix or the default namespace.

**Signature**

```java
public String getNamespace()
```

**Return Value**

Type: String

**Usage**

Returns null if the XML event does not have a prefix.
getNamespaceCount()
Returns the number of namespaces declared on a start element or end element.

Signature
public Integer getNamespaceCount()

Return Value
Type: Integer

Usage
This method is only valid on a start element, end element, or namespace XML event.

getNamespacePrefix(index)
Returns the prefix for the namespace declared at the index.

Signature
public String getNamespacePrefix(Integer index)

Parameters
index
  Type: Integer

Return Value
Type: String

Usage
Returns null if this is the default namespace declaration. This method is only valid on a start element, end element, or namespace XML event.

getNamespaceURI(prefix)
Return the URI for the given prefix.

Signature
public String getNamespaceURI(String prefix)

Parameters
prefix
  Type: String
Return Value
Type: String

Usage
The returned URI depends on the current state of the processor.

**getNamespaceURIAt(index)**
Returns the URI for the namespace declared at the index.

**Signature**
```java
public String getNamespaceURIAt(Integer index)
```

**Parameters**
- **index**
  Type: Integer

**Return Value**
Type: String

Usage
This method is only valid on a start element, end element, or namespace XML event.

**getPIData()**
Returns the data section of a processing instruction.

**Signature**
```java
public String getPIData()
```

**Return Value**
Type: String

**getPITarget()**
Returns the target section of a processing instruction.

**Signature**
```java
public String getPITarget()
```

**Return Value**
Type: String
getPrefix()  
Returns the prefix of the current XML event or null if the event does not have a prefix.

**Signature**  
```java
public String getPrefix()
```

**Return Value**  
Type: `String`

getText()  
Returns the current value of the XML event as a string.

**Signature**  
```java
public String getText()
```

**Return Value**  
Type: `String`

**Usage**  
The valid values for the different events are:
- The string value of a character XML event
- The string value of a comment
- The replacement value for an entity reference. For example, assume `getText` reads the following XML snippet:

```xml
<!ENTITY Title "Salesforce For Dummies" >
<moo a="b">Name &Title;</moo>'
```

The `getText` method returns `Salesforce for Dummies`, not `&Title`.
- The string value of a CDATA section
- The string value for a space XML event
- The string value of the internal subset of the DTD

getVersion()  
Returns the XML version specified on the XML declaration. Returns `null` if none was declared.

**Signature**  
```java
public String getVersion()
```
Return Value
Type: String

**hasName()**

Returns `true` if the current XML event has a name. Returns `false` otherwise.

**Signature**
`public Boolean hasName()`

**Return Value**
Type: Boolean

**Usage**
This method is only valid for start element and stop element XML events.

**hasNext()**

Returns `true` if there are more XML events and `false` if there are no more XML events.

**Signature**
`public Boolean hasNext()`

**Return Value**
Type: Boolean

**Usage**
This method returns `false` if the current XML event is end document.

**hasText()**

Returns `true` if the current event has text, `false` otherwise.

**Signature**
`public Boolean hasText()`

**Return Value**
Type: Boolean

**Usage**
The following XML events have text: characters, entity reference, comment and space.
isCharacters()
Returns true if the cursor points to a character data XML event. Otherwise, returns false.

Signature
public Boolean isCharacters()

Return Value
Type: Boolean

isEndElement()
Returns true if the cursor points to an end tag. Otherwise, it returns false.

Signature
public Boolean isEndElement()

Return Value
Type: Boolean

isStartElement()
Returns true if the cursor points to a start tag. Otherwise, it returns false.

Signature
public Boolean isStartElement()

Return Value
Type: Boolean

isWhiteSpace()
Returns true if the cursor points to a character data XML event that consists of all white space. Otherwise it returns false.

Signature
public Boolean isWhiteSpace()

Return Value
Type: Boolean
**next ()**
Reads the next XML event. A processor may return all contiguous character data in a single chunk, or it may split it into several chunks. Returns an integer which indicates the type of event.

**Signature**
`public Integer next()`

**Return Value**
Type: Integer

**nextTag ()**
Skips any white space (the `isWhiteSpace` method returns `true`), comment, or processing instruction XML events, until a start element or end element is reached. Returns the index for that XML event.

**Signature**
`public Integer nextTag()`

**Return Value**
Type: Integer

**Usage**
This method throws an error if elements other than white space, comments, processing instruction, start elements or stop elements are encountered.

**setCoalescing (returnAsSingleBlock)**
If you specify `true` for `returnAsSingleBlock`, text is returned in a single block, from a start element to the first end element or the next start element, whichever comes first. If you specify it as `false`, the parser may return text in multiple blocks.

**Signature**
`public Void setCoalescing(Boolean returnAsSingleBlock)`

**Parameters**
`returnAsSingleBlock`
Type: `Boolean`

**Return Value**
Type: Void
setNamespaceAware(isNamespaceAware)

If you specify true for isNamespaceAware, the parser recognizes namespace. If you specify it as false, the parser does not. The default value is true.

Signature

```java
public Void setNamespaceAware(Boolean isNamespaceAware)
```

Parameters

isNamespaceAware
Type: Boolean

Return Value

Type: Void

toString()

Returns a string containing the length of the input XML given to XmlStreamReader and the first 50 characters of the input XML.

Signature

```java
public String toString()
```

Return Value

Type: String

XmlStreamWriter Class

The XmlStreamWriter class provides methods for writing XML data.

Namespace

System

Usage

You can use the XmlStreamWriter class to programmatically construct an XML document, then use HTTP classes to send the document to an external server.

The XmlStreamWriter class is similar to the XMLStreamWriter utility class from StAX (Streaming API for XML). StAX is an API to read and write XML documents, originating from the Java programming language community.

Note: The XmlStreamWriter class in Apex is based on its counterpart in Java. See Java XMLStreamWriter class.

IN THIS SECTION:

XmlStreamWriter Constructors
XmlStreamWriter Methods

SEE ALSO:
- Http Class
- HttpRequest Class
- HttpResponse Class

XmlStreamWriter Constructors
The following are constructors for XmlStreamWriter.

IN THIS SECTION:
- XmlStreamWriter()
  Creates a new instance of the XmlStreamWriter class.

XmlStreamWriter()
Creates a new instance of the XmlStreamWriter class.

Signature

public XmlStreamWriter()

XmlStreamWriter Methods
The following are methods for XmlStreamWriter. All are instance methods.

IN THIS SECTION:
- close()
  Closes this instance of an XmlStreamWriter and free any resources associated with it.
- getXmlString()
  Returns the XML written by the XmlStreamWriter instance.
- setDefaultNamespace(uri)
  Binds the specified URI to the default namespace. This URI is bound in the scope of the current START_ELEMENT – END_ELEMENT pair.
- writeAttribute(prefix, namespaceUri, localName, value)
  Writes an attribute to the output stream.
- writeCData(data)
  Writes the specified CData to the output stream.
- writeCharacters(text)
  Writes the specified text to the output stream.
- writeComment(comment)
  Writes the specified comment to the output stream.
writeDefaultNamespace(namespaceUri)
Writes the specified namespace to the output stream.

writeEmptyElement(prefix, localName, namespaceUri)
Writes an empty element tag to the output stream.

writeEndElement()
Closes any start tags and writes corresponding end tags to the output stream.

writeEndDocument()
Writes an end tag to the output stream, relying on the internal state of the writer to determine the prefix and local name.

writeNamespace(prefix, namespaceUri)
Writes the specified namespace to the output stream.

writeProcessingInstruction(target, data)
Writes the specified processing instruction.

writeStartDocument(encoding, version)
Writes the XML Declaration using the specified XML encoding and version.

writeStartElement(prefix, localName, namespaceUri)
Writes the start tag specified by localName to the output stream.

close()
Closes this instance of an XmlStreamWriter and free any resources associated with it.

Signature
public Void close()

Return Value
Type: Void

getXmlString()
Returns the XML written by the XmlStreamWriter instance.

Signature
public String getXmlString()

Return Value
Type: String

setDefaultNamespace(uri)
Binds the specified URI to the default namespace. This URI is bound in the scope of the current START_ELEMENT – END_ELEMENT pair.
setDefaultNamespace(String uri)

Parameters
uri
Type: String

Return Value
Type: Void

writeAttribute(prefix, namespaceUri, localName, value)

Writes an attribute to the output stream.

writeCData(data)

Writers the specified CData to the output stream.
Return Value
Type: Void

**writeCharacters(text)**
Writes the specified text to the output stream.

**Signature**
```
public Void writeCharacters(String text)
```

**Parameters**
- **text**
  Type: `String`

**Return Value**
Type: Void

**writeComment(comment)**
Writes the specified comment to the output stream.

**Signature**
```
public Void writeComment(String comment)
```

**Parameters**
- **comment**
  Type: `String`

**writeDefaultNamespace(namespaceUri)**
Writes the specified namespace to the output stream.

**Signature**
```
public Void writeDefaultNamespace(String namespaceUri)
```

**Parameters**
- **namespaceUri**
  Type: `String`
**Return Value**
Type: Void

**writeEmptyElement**(prefix, localName, namespaceUri)
Writes an empty element tag to the output stream.

**Signature**
```java
public Void writeEmptyElement(String prefix, String localName, String namespaceUri)
```

**Parameters**
- **prefix**
  Type: String
- **localName**
  Type: String
  Specifies the name of the tag to be written.
- **namespaceUri**
  Type: String

**Return Value**
Type: Void

**writeEndDocument**()
Closes any start tags and writes corresponding end tags to the output stream.

**Signature**
```java
public Void writeEndDocument()
```

**Return Value**
Type: Void

**writeEndElement**()
Writes an end tag to the output stream, relying on the internal state of the writer to determine the prefix and local name.

**Signature**
```java
public Void writeEndElement()
```

**Return Value**
Type: Void
writeNamespace(prefix, namespaceUri)
Writes the specified namespace to the output stream.

Signature
public Void writeNamespace(String prefix, String namespaceUri)

Parameters
prefix
Type: String
namespaceUri
Type: String

Return Value
Type: Void

writeProcessingInstruction(target, data)
Writes the specified processing instruction.

Signature
public Void writeProcessingInstruction(String target, String data)

Parameters
target
Type: String
data
Type: String

Return Value
Type: Void

writeStartDocument(encoding, version)
Writes the XML Declaration using the specified XML encoding and version.

Signature
public Void writeStartDocument(String encoding, String version)

Parameters
encoding
Type: String
version
Type: String

Return Value
Type: Void

**writeStartElement(prefix, localName, namespaceUri)**
Writes the start tag specified by `localName` to the output stream.

**Signature**
```java
public Void writeStartElement(String prefix, String localName, String namespaceUri)
```

**Parameters**
- **prefix**
  Type: String
- **localName**
  Type: String
- **namespaceUri**
  Type: String

**Return Value**
Type: Void

**TerritoryMgmt Namespace**

The **TerritoryMgmt** namespace provides an interface used for territory management.
The following is the interface in the **TerritoryMgmt** namespace.

**IN THIS SECTION:**
- **OpportunityTerritory2AssignmentFilter Global Interface**
  Apex interface that allows an implementing class to assign a single territory to an opportunity.

**OpportunityTerritory2AssignmentFilter Global Interface**

Apex interface that allows an implementing class to assign a single territory to an opportunity.

**Namespace**

**TerritoryMgmt**
Usage

Method called by Opportunity Territory Assignment job to assign territory to opportunity. Input is a list of (up to 1000) opportunityIds that have IsExcludedFromTerritory2Filter=false. Returns a map of OpportunityId to Territory2Id, which is used to update the Territory2Id field on the Opportunity object.

OpportunityTerritory2AssignmentFilter Methods

The following are methods for OpportunityTerritory2AssignmentFilter.

getOpportunityTerritory2Assignments(opportunityIds)

Returns the mapping of opportunities to territory IDs. When Salesforce invokes this method, it supplies the list of opportunity IDs, except for opportunities that have been excluded from territory assignment (IsExcludedFromTerritory2Filter=false).

Signature

public Map<Id,Id> getOpportunityTerritory2Assignments(List<Id> opportunityIds)

Parameters

opportunityIds

Type: List<Id>
Opportunity IDs.

Return Value

Type: Map<Id,Id>
A key value pair associating each Territory ID to an Opportunity ID.

OpportunityTerritory2AssignmentFilter Example Implementation

This is an example implementation of the TerritoryMgmt.OpportunityTerritory2AssignmentFilter interface.

```apex
/**
 * Apex version of the default logic.
 * If opportunity's assigned account is assigned to
 * Case 1: 0 territories in active model
 * then set territory2Id = null
 * Case 2: 1 territory in active model
 */
```
then set territory2Id = account's territory2Id

Case 3: 2 or more territories in active model
then set territory2Id = account’s territory2Id that is of highest priority.
But if multiple territories have same highest priority, then set territory2Id = null

*/
global class OppTerrAssignDefaultLogicFilter implements TerritoryMgmt.OpportunityTerritory2AssignmentFilter {
/**
* No-arg constructor.
*/
global OppTerrAssignDefaultLogicFilter() {}

/**
* Get mapping of opportunity to territory2Id. The incoming list of opportunityIds contains only those with IsExcludedFromTerritory2Filter=false.
* If territory2Id = null in result map, clear the opportunity.territory2Id if set.
* If opportunity is not present in result map, its territory2Id remains intact.
*/
global Map<Id,Id> getOpportunityTerritory2Assignments(List<Id> opportunityIds) {
    Map<Id, Id> OppIdTerritoryIdResult = new Map<Id, Id>();

    // Get the active territory model Id
    Id activeModelId = getActiveModelId();

    if(activeModelId != null){
        List<Opportunity> opportunities =
            [Select Id, AccountId, Territory2Id from Opportunity where Id IN :opportunityIds];
        Set<Id> accountIds = new Set<Id>();
        // Create set of parent accountIds
        for(Opportunity opp:opportunities){
            if(opp.AccountId != null){
                accountIds.add(opp.AccountId);
            }
        }

        Map<Id,Territory2Priority> accountMaxPriorityTerritory =
            getAccountMaxPriorityTerritory(activeModelId, accountIds);

        // For each opportunity, assign the highest priority territory if there is no conflict, else assign null.
        for(Opportunity opp: opportunities){
            Territory2Priority tp = accountMaxPriorityTerritory.get(opp.AccountId);
            // Assign highest priority territory if there is only 1.
            if((tp != null) && (tp.moreTerritoriesAtPriority == false) && (tp.territory2Id != opp.Territory2Id)){
                OppIdTerritoryIdResult.put(opp.Id, tp.territory2Id);
            }else{
                OppIdTerritoryIdResult.put(opp.Id, null);
            }
        }
    }
    return OppIdTerritoryIdResult;
}
private Map<Id, Territory2Priority> getAccountMaxPriorityTerritory(Id activeModelId, Set<Id> accountIds){
    Map<Id, Territory2Priority> accountMaxPriorityTerritory = new Map<Id, Territory2Priority>();
    for(ObjectTerritory2Association ota:[Select ObjectId, Territory2Id, Territory2.Type.Priority from ObjectTerritory2Association where objectId IN :accountIds and Territory2.Type.ModelId = :activeModelId]){
        Territory2Priority tp = accountMaxPriorityTerritory.get(ota.ObjectId);
        if((tp == null) || (ota.Territory2.Type.Priority > tp.priority)){
            // If this is the first territory examined for account or it has greater priority than current highest priority territory, then set this as new highest priority territory.
            tp = new Territory2Priority(ota.Territory2Id, ota.Territory2.Type.Priority, false);
        } else if(ota.Territory2.Type.priority == tp.priority){
            // The priority of current highest territory is same as this, so set moreTerritoriesAtPriority to indicate multiple highest priority territories seen so far.
            tp.moreTerritoriesAtPriority = true;
        }
        accountMaxPriorityTerritory.put(ota.ObjectId, tp);
    }
    return accountMaxPriorityTerritory;
}

/**
 * Get the Id of the Active Territory Model.
 * If none exists, return null.
 */
private Id getActiveModelId() {
    List<Territory2Model> models = [Select Id from Territory2Model where State = 'Active'];
    Id activeModelId = null;
    if(models.size() == 1){
        activeModelId = models.get(0).Id;
    }
    return activeModelId;
}

/**
 * Helper class to help capture territory2Id, its priority, and whether there are more territories with same priority assigned to the account.
 */
private class Territory2Priority {
    public Id territory2Id { get; set; }
public Integer priority { get; set; }
public Boolean moreTerritoriesAtPriority { get; set; }

Territory2Priority(Id territory2Id, Integer priority, Boolean moreTerritoriesAtPriority)
{
    this.territory2Id = territory2Id;
    this.priority = priority;
    this.moreTerritoriesAtPriority = moreTerritoriesAtPriority;
}
}

TxnSecurity Namespace

The TxnSecurity namespace provides an interface used for transaction security.
The following is the interface and its supporting class in the TxnSecurity namespace.

IN THIS SECTION:

**Event Class**
Contains event information that the `evaluate` method uses to evaluate a transaction security policy.

**EventCondition Interface**
Allows an implementing class to specify whether to take action when certain events occur based on a transaction security policy. This interface is only used for Apex policies created in Real-Time Event Monitoring.

**AsyncCondition Interface**
Allows an implementing class to make asynchronous Apex calls. This interface is used only for transaction security Apex policies created in Real-Time Event Monitoring.

**PolicyCondition Interface**
Apex interface that allows an implementing class to specify actions to take when certain events occur based on a transaction security policy.

**Event Class**
Contains event information that the `evaluate` method uses to evaluate a transaction security policy.

**Namespace**
 TxnSecurity

**Usage**
The Event class contains the information needed to determine if the event triggers a Transaction Security policy. Not all class attributes are used for every type of event.

⚠️ Tip: The `EventClass` interface applies only to Legacy Transaction Security, which is a retired feature as of Summer '20. Use the `EventCondition` interface instead of the `EventClass` interface.
Event Constructors

The following is the constructor for Event.

**Event()**

Creates an instance of the TxnSecurity.Event class.

**Signature**

```java
public Event()
```

Event Properties

The following are properties for Event.

**IN THIS SECTION:**

- `action`
  Specifies the action being taken on the resource for an Entity event. For example, a Login IP resource for an Entity event could have an action of `create`. The `action` attribute is not used by any other event type.

- `data`
  Contains data used by actions. For example, `data` for a login event includes the login history ID. Returns a map whose keys are the type of event data, like `SourceIp`.

- `entityId`
  The ID of any entity associated with the event. For example, the `entityId` of a DataExport event for an Account object contains the Account ID.

- `entityName`
  The name of the object the event acts on.

- `organizationId`
  The ID of the Salesforce org where the event occurred.

- `resourceType`
  The type of resource for the event. For example, an AccessResource event could have a Connected Application as a resource type. Not all event types have resources.

- `timeStamp`
  The time the event occurred.
userId
Identifies the user that caused the event.

**action**

Specifies the action being taken on the resource for an Entity event. For example, a Login IP resource for an Entity event could have an action of create. The action attribute is not used by any other event type.

**Signature**

```java
public String action {get; set;}
```

**Property Value**

Type: String

**data**

Contains data used by actions. For example, data for a login event includes the login history ID. Returns a map whose keys are the type of event data, like SourceIp.

**Signature**

```java
public Map<String,String> data {get; set;}
```

**Property Value**

Type: Map<String, String>

The following table lists all the available data types. Not all types appear with all event types. The data type values are always string representations. For example, the `isApi` value is a string in the map, but is actually a Boolean value. Convert the value from a string to its true type this way: `Boolean.valueOf(event.data.get('isApi'));`

<table>
<thead>
<tr>
<th>Key Name</th>
<th>True Value Type</th>
<th>Events Supported</th>
</tr>
</thead>
<tbody>
<tr>
<td>ActionName</td>
<td><strong>String</strong> Values are:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Convert</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Delete</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Insert</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Undelete</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Update</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Upsert</td>
<td></td>
</tr>
<tr>
<td>ApiType</td>
<td><strong>String</strong> (Enum manifested as a String)</td>
<td>DataExport, Login</td>
</tr>
<tr>
<td>Application</td>
<td><strong>String</strong></td>
<td></td>
</tr>
<tr>
<td>ClientId</td>
<td><strong>String</strong> (ID of the client)</td>
<td>DataExport</td>
</tr>
<tr>
<td>ConnectedAppId</td>
<td><strong>String</strong> (ID of the Connected App)</td>
<td>AccessResource, DataExport</td>
</tr>
<tr>
<td>Key Name</td>
<td>True Value Type</td>
<td>Events Supported</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>ExecutionTime</td>
<td>milliseconds</td>
<td>DataExport</td>
</tr>
<tr>
<td>IsApi</td>
<td>Boolean</td>
<td>DataExport</td>
</tr>
<tr>
<td>IsScheduled</td>
<td>Boolean</td>
<td>DataExport</td>
</tr>
<tr>
<td>LoginHistoryId</td>
<td>String</td>
<td>DataExport, Login</td>
</tr>
<tr>
<td>NumberOfRecords</td>
<td>Integer</td>
<td>DataExport</td>
</tr>
<tr>
<td>PolicyId</td>
<td>String (ID of the current policy)</td>
<td>All events</td>
</tr>
<tr>
<td>SessionLevel</td>
<td>String (Enum manifested as a String. Values include STANDARD and HIGH_ASSURANCE)</td>
<td>AccessResource</td>
</tr>
<tr>
<td>SourceIp</td>
<td>String (IPV4 Address)</td>
<td>AccessResource</td>
</tr>
<tr>
<td>UserName</td>
<td>String</td>
<td>Entity</td>
</tr>
</tbody>
</table>

**entityId**

The ID of any entity associated with the event. For example, the entityId of a DataExport event for an Account object contains the Account ID.

**Signature**

```java
public String entityId {get; set;}
```

**Property Value**

Type: String

**entityName**

The name of the object the event acts on.

**Signature**

```java
public String entityName {get; set;}
```

**Property Value**

Type: String

**organizationId**

The ID of the Salesforce org where the event occurred.

**Signature**

```java
public String organizationId {get; set;}
```
Property Value
Type: String

**resourceType**
The type of resource for the event. For example, an AccessResource event could have a Connected Application as a resource type. Not all event types have resources.

Signature

```java
public String resourceType {get; set;}
```

Property Value
Type: String

**timeStamp**
The time the event occurred.

Signature

```java
public Datetime timeStamp {get; set;}
```

Property Value
Type: Datetime

**userId**
Identifies the user that caused the event.

Signature

```java
public String userId {get; set;}
```

EventCondition Interface

Allows an implementing class to specify whether to take action when certain events occur based on a transaction security policy. This interface is only used for Apex policies created in Real-Time Event Monitoring.

Usage

The `evaluate` method is called upon the occurrence of a real-time event monitored by a transaction security policy. A typical implementation first selects the fields of interest from the event. Then the fields are tested to see if they meet the conditions being monitored. If the conditions are met, the method returns `true`. 
For example, imagine a transaction security policy that triggers when a user queries more than 1,000 lead records. For each API event, the `evaluate` method checks whether the `RowsProcessed` value is greater than 1,000 and the `QueriedEntities` value contains “Lead”. If so, `true` is returned.

We recommend having test classes for the policy condition interface to ensure it works correctly. Testing is required regardless of whether the policy is moved from a sandbox to production, with a change set, or some other way. For example, test your policies in your development environment before moving the policies to production.

For more information about testing Apex transaction security policies, read Transaction Security Apex Testing.

### EventCondition Methods

The following are methods for `EventCondition`.

**IN THIS SECTION:**
- EventCondition Methods
- EventCondition Example Implementation

#### EventCondition Methods

The following are methods for `EventCondition`.

**IN THIS SECTION:**
- `evaluate(event)`

#### `evaluate(event)`

Evaluates an event against a transaction security policy created in Real-Time Event Monitoring. If the event triggers the policy, the method returns `true`.

**Signature**

```
public Boolean evaluate(SObject event)
```

**Parameters**

- `var1`
  - Type: `SObject`
  - The event to check against the transaction security policy.

**Return Value**

- Type: `Boolean`
  - Returns `true` when the policy is triggered. For example, suppose that the policy is to limit users to a single login session. If a user tries to log in a second time, the policy blocks the attempted login, and updates the `Status`, `PolicyId`, and `PolicyOutcome` fields of that `LoginEvent`. The policy also sends an email notification to the Salesforce admin. The `evaluate` method only checks the login event, and returns `true` if it’s the user’s second login attempt.
  - The system performs the action and notification, not the `evaluate` method.
EventCondition Example Implementation

This example shows an implementation of the TxnSecurity.EventCondition interface. The transaction security policy triggers when the user queries an Account object.

```
global class BlockAccountQueriesEventCondition implements TxnSecurity.EventCondition {

    public boolean evaluate(SObject event) {
        switch on event {
            when ApiEvent apiEvent {
                return handleApiEvent(apiEvent);
            }
            when null {
                // Trigger action if event is null
                return true;
            }
            when else {
                // Trigger action for unhandled events
                return true;
            }
        }
    }

    private boolean handleApiEvent(ApiEvent apiEvent) {
        if(apiEvent.QueriedEntities.contains('Account')){
            return true;
        }
        return false;
    }
}
```

For more examples, see Enhanced Apex Transaction Security Implementation Examples.

AsyncCondition Interface

Allows an implementing class to make asynchronous Apex calls. This interface is used only for transaction security Apex policies created in Real-Time Event Monitoring.

Namespace

TxnSecurity

Usage

If you make an Asynchronous Apex call in the class that implements your transaction security policy condition, the class must implement the TxnSecurity.AsyncCondition interface in addition to TxnSecurity.EventCondition. Use Asynchronous Apex instead of Apex callouts and DML statements, neither of which is allowed in transaction security Apex policies.

Apex offers multiple ways to run your Apex code asynchronously and all are supported in the TxnSecurity.AsyncCondition interface.

This interface has no methods.
AsyncCondition Example Implementation

Here's an example implementation of the `TxnSecurity.AsyncCondition` interface. The transaction security policy triggers when a user logs in. In the example, `ExternalValidation__c` is a custom object that contains information from an external validation system. The result of the SOQL query on `ExternalValidation__c` determines whether to block the user from logging in. The policy then queues the `CalloutToExternalValidator` class for asynchronous execution. When it executes, the `CalloutToExternalValidator` class makes an external call to the validation system to update it with information about this log in event. Because `CalloutToExternalValidator` is triggered by Asynchronous Apex, you must implement the `TxnSecurity.AsyncCondition` interface in the `ExternallyValidatedLoginCondition` Apex class along with the usual `TxnSecurity.EventCondition` interface.

```java
global class ExternallyValidatedLoginCondition implements TxnSecurity.EventCondition,
    TxnSecurity.AsyncCondition {
    public boolean evaluate(SObject event) {
        LoginEvent loginEvent = (LoginEvent) event;
        Boolean userBlocked = [select blocked from ExternalValidation__c where loginId =
            loginEvent.UserId][0].Blocked;

        System.enqueueJob(new CalloutToExternalValidator(loginEvent.SourceIp,
            loginEvent.LoginUrl));
        return userBlocked;
    }
}

public class CalloutToExternalValidator implements Queueable {
    private String sourceIp;
    private String loginUrl;

    public CalloutToExternalValidator(String sourceIp, String loginUrl) {
        this.sourceIp = sourceIp;
        this.loginUrl = loginUrl;
    }

    public void execute(QueueableContext context) {
        // callout to external validation service
        // pass sourceIp, loginUrl

        // update ExternalValidation__c
    }
}
```

PolicyCondition Interface

Apex interface that allows an implementing class to specify actions to take when certain events occur based on a transaction security policy.
Namespace

`TxnSecurity`

Usage

⚠️ **Tip:** The PolicyCondition interface applies only to Legacy Transaction Security, which is a retired feature as of Summer '20. Use the EventCondition interface instead of the PolicyCondition interface.

The `evaluate` method is called upon the occurrence of an event monitored by a transaction security policy. A typical implementation first selects the item of interest from the event. Then the item is tested to see if it meets the condition being monitored. If the condition is met, the method returns `true`.

For example, imagine a transaction security policy that checks for the same user logging in more than once. For each login event, the method would check if the user logging in already has a login session in progress, and if so, `true` is returned.

We recommend having test classes for the policy condition interface to ensure it works correctly. Testing is required regardless of whether the policy is moved from a sandbox to production, with a change set, or some other way. For example, test your policies in your development environment before moving the policies to production.

Don't include DML statements in your custom policies because they can cause errors. When you send a custom email via Apex during transaction policy evaluation, you get an error, even if the record isn't explicitly related to another record. For more information, see Apex DML Operations in the Apex Reference Guide.

IN THIS SECTION:

PolicyCondition Methods

PolicyCondition Methods

The following is the method for PolicyCondition.

IN THIS SECTION:

`evaluate(event)`

Evaluates an event against a transaction security policy. If the event triggers the policy, `true` is returned.

**Signature**

```java
public Boolean evaluate(TxnSecurity.Event event)
```

**Parameters**

`event`

Type: `TxnSecurity.Event`

The event to check against the transaction security policy.
Return Value

Type: Boolean

When the policy is triggered, True is returned. For example, let’s suppose the policy is to limit users to a single login session. If anyone tries to log in a second time, the policy’s action requires that they end their current session. The policy also sends an email notification to the Salesforce admin. The evaluate() method only checks the login event, and returns True if it’s the user’s second login. The Transaction Security system performs the action and notification, and not the evaluate() method.

UserProvisioning Namespace

The UserProvisioning namespace provides methods for monitoring outbound user provisioning requests.

The following is the class in the UserProvisioning namespace.

IN THIS SECTION:

ConnectorTestUtil Class
Enables developers to write Apex test classes for connectors used by the connected app provisioning solution. This class simulates provisioning for the associated app.

UserProvisioningLog Class
Provides methods for writing messages to monitor outbound user provisioning requests.

UserProvisioningPlugin Class
The UserProvisioningPlugin base class implements Process.Plugin for programmatic customization of the user provisioning process for connected apps.

ConnectorTestUtil Class

Enables developers to write Apex test classes for connectors used by the connected app provisioning solution. This class simulates provisioning for the associated app.

Namespace

UserProvisioning

Usage

Use this class for connector-based test accelerators. You can invoke it only from within an Apex test.

Example

This example creates an instance of a connected app, gets a value, and checks whether the value is correct. The test is simply a row inserted in the database table.

```apex
@isTest
private class SCIMCreateUserPluginTest {
    public static void callPlugin(Boolean validInputParams) {
        // Code goes here
    }
```
//Create an instance of a connected app
ConnectedApplication capp = UserProvisioning.ConnectorTestUtil.createConnectedApp('TestApp');
Profile p = [SELECT Id FROM Profile WHERE Name='Standard User'];
//Create a user
User user = new User(username='testuser1@scimuserprov.test', FirstName='Test',
LastName='User1', email='testuser1@testemail.com', FederationIdentifier='testuser1@testemail.com', profileId= p.Id,
communityNickName='tuser1', alias='tuser', TimeZoneSidKey='GMT',
LocaleSidKey='en_US', EmailEncodingKey='ISO-8859-1', LanguageLocaleKey='en_US');
//insert user into a row in the database table
 insert user;
//Create a UPR
UserProvisioningRequest upr = new UserProvisioningRequest(appname = capp.name,
connectedAppId=capp.id, operation='Create',
state='New', approvalStatus='NotRequired',salesforceUserId=user.id);
//Insert the UPR to test the flow end to end
 insert upr;
}}

IN THIS SECTION:
ConnectorTestUtil Method

SEE ALSO:
Salesforce Help: User Provisioning for Connected Apps

ConnectorTestUtil Method

The ConnectorTestUtil class has 1 method.

IN THIS SECTION:
createConnectedApp(connectedAppName)
Creates an instance of a connected app to simulate provisioning.

createConnectedApp (connectedAppName)
Creates an instance of a connected app to simulate provisioning.

Signature
public static ConnectedApplication createConnectedApp(String connectedAppName)

Parameters
connectedAppName
Type: String
Name of the connected app to test for provisioning.

Return Value
Type: ConnectedApplication
The instance of the connected app to test for provisioning.

UserProvisioningLog Class
Provides methods for writing messages to monitor outbound user provisioning requests.

Namespace
UserProvisioning

Example
This example writes the user account information sent to a third-party system for a provisioning request to the UserProvisioningLog object.

```java
String inputParamsStr = 'Input parameters: uprId=' + uprId + ',
endpointURL=' + endpointURL + ', adminUsername=' + adminUsername + ',
email=' + email + ', username=' + username + ', defaultPassword=' + defaultPassword + ',
defaultRoles = ' + defaultRoles;
UserProvisioning.UserProvisioningLog.log(uprId, inputParamsStr);
```

IN THIS SECTION:
UserProvisioningLog Methods

UserProvisioningLog Methods
The following are methods for UserProvisioningLog. All methods are static.

IN THIS SECTION:
- `log(userProvisioningRequestId, details)`
  Writes a specific message, such as an error message, to monitor the progress of a user provisioning request.
- `log(userProvisioningRequestId, status, details)`
  Writes a specific status and message, such as a status and detailed error message, to monitor the progress of a user provisioning request.
- `log(userProvisioningRequestId, externalUserId, externalUserName, userId, details)`
  Writes a specific message, such as an error message, to monitor the progress of a user provisioning request associated with a specific user.
- `log(userProvisioningRequestId, details)`
  Writes a specific message, such as an error message, to monitor the progress of a user provisioning request.
Signature

```java
public void log(String userProvisioningRequestId, String details)
```

Parameters

- `userProvisioningRequestId`
  - Type: `String`
  - A unique identifier for the user provisioning request.

- `details`
  - Type: `String`
  - The text for the message.

Return Value

Type: `void`

### log(userProvisioningRequestId, status, details)

Writes a specific status and message, such as a status and detailed error message, to monitor the progress of a user provisioning request.

Signature

```java
public void log(String userProvisioningRequestId, String status, String details)
```

Parameters

- `userProvisioningRequestId`
  - Type: `String`
  - A unique identifier for the user provisioning request.

- `status`
  - Type: `String`
  - A description of the current state. For example, while invoking a third-party API, the status could be `invoke`.

- `details`
  - Type: `String`
  - The text for the message.

Return Value

Type: `void`

### log(userProvisioningRequestId, externalUserId, externalUserName, userId, details)

Writes a specific message, such as an error message, to monitor the progress of a user provisioning request associated with a specific user.
Signature

```java
public void log(String userProvisioningRequestId, String externalUserId, String externalUserName, String userId, String details)
```

**Parameters**

- `userProvisioningRequestId`
  - Type: `String`
  - A unique identifier for the user provisioning request.

- `externalUserId`
  - Type: `String`
  - The unique identifier for the user in the target system.

- `externalUserName`
  - Type: `String`
  - The username for the user in the target system.

- `userId`
  - Type: `String`
  - Salesforce ID of the user making the request.

- `details`
  - Type: `String`
  - The text for the message.

**Return Value**

Type: `void`

**UserProvisioningPlugin Class**

The `UserProvisioningPlugin` base class implements `Process.Plugin` for programmatic customization of the user provisioning process for connected apps.

**Namespace**

`UserProvisioning`

**Usage**

Extending this class gives you a plug-in that can be used Flow Builder as a legacy Apex action, with the following input and output parameters.

<table>
<thead>
<tr>
<th>Input Parameter Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>userProvisioningRequestId</code></td>
<td>The unique ID of the request for the plug-in to process.</td>
</tr>
<tr>
<td><code>userId</code></td>
<td>The ID of the associated user for the request.</td>
</tr>
</tbody>
</table>
### Input Parameter Name

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>NamedCredDevName</strong></td>
</tr>
<tr>
<td><strong>reconFilter</strong></td>
</tr>
<tr>
<td><strong>reconOffset</strong></td>
</tr>
</tbody>
</table>

### Output Parameter Name

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Status</strong></td>
</tr>
<tr>
<td><strong>Details</strong></td>
</tr>
<tr>
<td><strong>ExternalUserId</strong></td>
</tr>
<tr>
<td><strong>ExternalUsername</strong></td>
</tr>
<tr>
<td><strong>ExternalEmail</strong></td>
</tr>
<tr>
<td><strong>ExternalFirstName</strong></td>
</tr>
<tr>
<td><strong>ExternalLastName</strong></td>
</tr>
<tr>
<td><strong>reconState</strong></td>
</tr>
<tr>
<td><strong>nextReconOffset</strong></td>
</tr>
</tbody>
</table>

If you want to add more custom parameters, use the `buildDescribeCall()` method.
**Example**

The following example uses the `buildDescribeCall()` method to add a new input parameter and a new output parameter. The example also demonstrates how to bypass the limit of the 10,000 records processed in DML statements in an Apex transaction.

```apex
global class SampleConnector extends UserProvisioning.UserProvisioningPlugin {

    // Example of adding more input and output parameters to those defined in the base class
    global override Process.PluginDescribeResult buildDescribeCall() {
        Process.PluginDescribeResult describeResult = new Process.PluginDescribeResult();
        describeResult.inputParameters = new List<Process.PluginDescribeResult.InputParameter>{
        };
        describeResult.outputParameters = new List<Process.PluginDescribeResult.OutputParameter>{
        };
        return describeResult;
    }

    // Example Plugin that demonstrates how to leverage the reconOffset/nextReconOffset/reconState
    // parameters to create more than 10,000 users. (i.e. go beyond the 10,000 DML limit per transaction)
    global override Process.PluginResult invoke(Process.PluginRequest request) {
        Map<String,String> result = new Map<String,String>();
        String uprId = (String) request.inputParameters.get('userProvisioningRequestId');
        UserProvisioning.UserProvisioningLog.log(uprId, 'Inserting Log from test Apex connector');
        UserProvisioningRequest upr = [SELECT id, operation, connectedAppId, state
            FROM userprovisioningrequest WHERE id = :uprId];
        if (upr.operation.equals('Reconcile')) {
            String reconOffsetStr = (String) request.inputParameters.get('reconOffset');
            Integer reconOffset = 0;
            if (reconOffsetStr != null) {
                reconOffset = Integer.valueOf(reconOffsetStr);
            }
            if (reconOffset > 44999) {
                result.put('reconState', 'Completed');
            }
            Integer i = 0;
            List<UserProvAccountStaging> upasList = new List<UserProvAccountStaging>();
            for (i = 0; i < 5000; i++) {
                UserProvAccountStaging upas = new UserProvAccountStaging();
            }
        }
    }
}
```
upas.Name = i + reconOffset + '';
upas.ExternalFirstName = upas.Name;
upas.ExternalEmail = 'externaluser@externalsystem.com';
upas.LinkState = 'Orphaned';
upas.Status = 'Active';
upas.connectedAppId = upr.connectedAppId;
upasList.add(upas);
}
insert upasList;
result.put('nextReconOffset', reconOffset + 5000 + '');
}
return new Process.PluginResult(result);
}

IN THIS SECTION:
UserProvisioningPlugin Methods

UserProvisioningPlugin Methods

The following are methods for UserProvisioningPlugin.

IN THIS SECTION:
buildDescribeCall()
Use this method to add more input and output parameters to those defined in the base class.
describe()
Returns a Process.PluginDescribeResult object that describes this method call.
getPluginClassName()
Returns the name of the class implementing the plugin.
invoke(request)
Primary method that the system invokes when the class that implements the interface is instantiated.

buildDescribeCall()

Use this method to add more input and output parameters to those defined in the base class.

Signature

public Process.PluginDescribeResult buildDescribeCall()

Return Value

Type: Process.PluginDescribeResult
describe()  
Returns a Process.PluginDescribeResult object that describes this method call.

**Signature**  
```java
public Process.PluginDescribeResult describe()
```

**Return Value**  
Type: Process.PluginDescribeResult

getPluginClassName()  
Returns the name of the class implementing the plugin.

**Signature**  
```java
public String getPluginClassName()
```

**Return Value**  
Type: String

invoke(request)  
Primary method that the system invokes when the class that implements the interface is instantiated.

**Signature**  
```java
```

**Parameters**  
```
request  
Type: Process.PluginRequest
```

**Return Value**  
Type: Process.PluginDescribeResult

---

**VisualEditor Namespace**

The **VisualEditor** namespace provides classes and methods for interacting with the Lightning App Builder. The classes and methods in this namespace operate on Lightning components, which include Lightning web components and Aura components. As of Spring ’19 (API version 45.0), you can build Lightning components using two programming models: the Lightning Web Components model, and the original Aura Components model. Lightning web components are custom HTML elements built using HTML and modern JavaScript. Lightning web components and Aura components can coexist and interoperate on a page.
Configure Lightning web components and Aura components to work in Lightning App Builder and Experience Builder. Admins and end users don’t know which programming model was used to develop the components. To them, they’re simply Lightning components.

The following are the classes in the VisualEditor namespace.

IN THIS SECTION:
- **DataRow Class**
  Contains information about one item in a picklist used in a Lightning component on a Lightning page.
- **DesignTimePageContext Class**
  A class that provides context information about a Lightning page. It can be used to help define the values of a picklist in a Lightning component on a Lightning page based on the page’s type and the object with which it’s associated.
- **DynamicPickList Class**
  An abstract class, used to display the values of a picklist in a Lightning component on a Lightning page.
- **DynamicPickListRows Class**
  Contains a list of picklist items in a Lightning component on a Lightning page.

**DataRow Class**

Contains information about one item in a picklist used in a Lightning component on a Lightning page.

**Namespace**

VisualEditor

IN THIS SECTION:
- **DataRow Constructors**
- **DataRow Methods**

**DataRow Constructors**

The following are constructors for DataRow.

IN THIS SECTION:
- **DataRow(label, value, selected)**
  Creates an instance of the VisualEditor.DataRow class using the specified label, value, and selected option.
- **DataRow(label, value)**
  Creates an instance of the VisualEditor.DataRow class using the specified label and value.

**DataRow(label, value, selected)**

Creates an instance of the VisualEditor.DataRow class using the specified label, value, and selected option.

**Signature**

```java
public DataRow(String label, Object value, Boolean selected)
```
Parameters

**label**
Type: String
User-facing label for the picklist item.

**value**
Type: Object
The value of the picklist item.

**selected**
Type: Boolean
Specifies whether the picklist item is selected (true) or not (false).

**DataRow(label, value)**
Creates an instance of the `VisualEditor.DataRow` class using the specified label and value.

**Signature**

```java
public DataRow(String label, Object value)
```

**Parameters**

**label**
Type: String
User-facing label for the picklist item.

**value**
Type: Object
The value of the picklist item.

**DataRow Methods**
The following are methods for `DataRow`.

IN THIS SECTION:

- **clone()**
  Makes a duplicate copy of the `VisualEditor.DataRow` object.

- **compareTo(o)**
  Compares the current `VisualEditor.DataRow` object to the specified one. Returns an integer value that is the result of the comparison.

- **getLabel()**
  Returns the user-facing label of the picklist item.

- **getValue()**
  Returns the value of the picklist item.

- **isSelected()**
  Returns the state of the picklist item, indicating whether it's selected or not.
clone ()
Makes a duplicate copy of the VisualEditor.DataRow object.

Signature
public Object clone()  

Return Value
Type: Object

compareTo (o)
Compares the current VisualEditor.DataRow object to the specified one. Returns an integer value that is the result of the comparison.

Signature
public Integer compareTo(VisualEditor.DataRow o)

Parameters
- o  
  Type: VisualEditor.DataRow
  A single item in a picklist.

Return Value
Type: Integer
Returns one of the following values:
- Zero if the current package version is equal to the specified package version
- An integer value greater than zero if the current package version is greater than the specified package version
- An integer value less than zero if the current package version is less than the specified package version

getLabel ()
Returns the user-facing label of the picklist item.

Signature
public String getLabel()

Return Value
Type: String

getValue ()
Returns the value of the picklist item.
Signature

```java
public Object getValue()
```

Return Value
Type: Object

**isSelected()**
Returns the state of the picklist item, indicating whether it’s selected or not.

Signature

```java
public Boolean isSelected()
```

Return Value
Type: Boolean

### DesignTimePageContext Class
A class that provides context information about a Lightning page. It can be used to help define the values of a picklist in a Lightning component on a Lightning page based on the page’s type and the object with which it’s associated.

### Namespace
VisualEditor

### Usage
To use this class, create a parameterized constructor in the custom Apex class that extends VisualEditor.DynamicPickList.

### Example
Here’s an example of a custom Apex class extending the VisualEditor.DynamicPickList class. It includes VisualEditor.DesignTimePageContext to define a picklist value that is available only if the page type is HomePage.

```java
global class MyCustomPickList extends VisualEditor.DynamicPickList{
    VisualEditor.DesignTimePageContext context;

    global MyCustomPickList(VisualEditor.DesignTimePageContext context) {
        this.context = context;
    }

    global override VisualEditor.DataRow getDefaultValue(){
        VisualEditor.DataRow defaultValue = new VisualEditor.DataRow('red', 'RED');
        return defaultValue;
    }

    global override VisualEditor.DynamicPickListRows getValues() {
        VisualEditor.DataRow value1 = new VisualEditor.DataRow('red', 'RED');
    }
}
```
VisualEditor.DataRow value2 = new VisualEditor.DataRow('yellow', 'YELLOW');
VisualEditor.DynamicPickListRows myValues = new VisualEditor.DynamicPickListRows();
myValues.addRow(value1);
myValues.addRow(value2);
if (context.pageType == 'HomePage') {
    VisualEditor.DataRow value3 = new VisualEditor.DataRow('purple', 'PURPLE');
    myValues.addRow(value3);
}
return myValues;
}

DesignTimePageContext Class

IN THIS SECTION:
DesignTimePageContext Properties
DesignTimePageContext Methods

DesignTimePageContext Properties

The following are properties for DesignTimePageContext.

IN THIS SECTION:

entityName
The API name of the sObject that a Lightning page is associated with, such as Account, Contact, or Custom_object__c. entityType
is available only for object pages, and not all Lightning pages are associated with objects.

pageType
The type of Lightning page, such as HomePage, AppPage, or RecordPage.

entityName

The API name of the sObject that a Lightning page is associated with, such as Account, Contact, or Custom_object__c. entityType
is available only for object pages, and not all Lightning pages are associated with objects.

Signature

public String entityId {get; set;}

Property Value

Type: String

pageType

The type of Lightning page, such as HomePage, AppPage, or RecordPage.
Signature

public String pageType {get; set;}

Property Value

Type: String

DesignTimePageContext Methods

The following are methods for DesignTimePageContext.

IN THIS SECTION:

clone()  

clone()  

Signature

public Object clone()

Return Value

Type: Object

DynamicPickList Class

An abstract class, used to display the values of a picklist in a Lightning component on a Lightning page.

Namespace

VisualEditor

Usage

To use this class as the datasource of a picklist in a Lightning component, it must be extended by a custom Apex class and then that class must be called in the component's design file.

Example

Here's an example of a custom Apex class extending the VisualEditor.DynamicPickList class.

global class MyCustomPickList extends VisualEditor.DynamicPickList{
    
    global override VisualEditor.DataRow getDefaultValue(){
        VisualEditor.DataRow defaultValue = new VisualEditor.DataRow('red', 'RED');
    
}
Here's an example of how the custom Apex class gets called in a design file so that the picklist appears in the Lightning component.

```xml
<design:component>
  <design:attribute name="property1" datasource="apex://MyCustomPickList"/>
</design:component>
```

IN THIS SECTION:
DynamicPickList Methods

DynamicPickList Methods

The following are methods for DynamicPickList.

IN THIS SECTION:
clone()  

global override VisualEditor.DynamicPicklistRows getValues() {
  VisualEditor.DataRow value1 = new VisualEditor.DataRow('red', 'RED');
  VisualEditor.DataRow value2 = new VisualEditor.DataRow('yellow', 'YELLOW');
  VisualEditor.DynamicPicklistRows myValues = new VisualEditor.DynamicPicklistRows();
  myValues.addRow(value1);
  myValues.addRow(value2);
  return myValues;
}

getDefaultValue()  
Returns the picklist item that is set as the default value for the picklist.

getLabel(attributeValue)  
Returns the user-facing label for a specified picklist value.

getValues()  
Returns the list of picklist item values.

isValid(attributeValue)  
Returns the valid state of the picklist item’s value. A picklist value is considered valid if it’s a part of any VisualEditor.DataRow in the VisualEditor.DynamicPicklistRows returned by getValues().

### clone()


**Signature**

```java
public Object clone()
```
Return Value
Type: Object

getDefaultValue()
Returns the picklist item that is set as the default value for the picklist.

Signature
public VisualEditor.DataRow getDefaultValue()

Return Value
Type: VisualEditor.DataRow

getLabel(attributeValue)
Returns the user-facing label for a specified picklist value.

Signature
public String getLabel(Object attributeValue)

Parameters
attributeValue
   Type: Object
   The value of the picklist item.

Return Value
Type: String

getValues()
Returns the list of picklist item values.

Signature
public VisualEditor.DynamicPickListRows getValues()

Return Value
Type: VisualEditor.DynamicPickListRows

isValid(attributeValue)
Returns the valid state of the picklist item's value. A picklist value is considered valid if it's a part of any VisualEditor.DataRow in the VisualEditor.DynamicPickListRows returned by getValues().
Signature

```java
public Boolean isValid(Object attributeValue)
```

Parameters

`attributeValue`

- **Type**: Object
- The value of the picklist item.

Return Value

- **Type**: Boolean

DynamicPickListRows Class

Contains a list of picklist items in a Lightning component on a Lightning page.

Namespace

`VisualEditor`

IN THIS SECTION:

- DynamicPickListRows Constructors
- DynamicPickListRows Methods

DynamicPickListRows Constructors

The following are constructors for DynamicPickListRows.

IN THIS SECTION:

- DynamicPickListRows(rows, containsAllRows)
- DynamicPickListRows(rows)
- DynamicPickListRows()

DynamicPickListRows(rows, containsAllRows)

- Creates an instance of the `VisualEditor.DynamicPickListRows` class using the specified parameters.

DynamicPickListRows(rows)

- Creates an instance of the `VisualEditor.DynamicPickListRows` class using the specified parameter.

DynamicPickListRows()

- Creates an instance of the `VisualEditor.DynamicPickListRows` class. You can then add rows by using the class's `addRow` or `addAllRows` methods.

**DynamicPickListRows(rows, containsAllRows)**

- Creates an instance of the `VisualEditor.DynamicPickListRows` class using the specified parameters.

**Signature**

```java
public DynamicPickListRows(List<VisualEditor.DataRow> rows, Boolean containsAllRows)
```
Parameters

rows
  Type: List VisualEditor.DataRow
  List of picklist items.

containsAllRows
  Type: Boolean
  Indicates if all values of the picklist are included in a type-ahead search query (true) or only those values initially displayed when the list is clicked on (false).

A picklist in a Lightning component can display only the first 200 values of a list. If containsAllRows is set to false, when a user does a type-ahead search to find values in the picklist, the search will only look at those first 200 values that were displayed, not the complete set of picklist values.

DynamicPickListRows(rows)

Creates an instance of the VisualEditor.DynamicPickListRows class using the specified parameter.

Signature

public DynamicPickListRows(List<VisualEditor.DataRow> rows)

Parameters

rows
  Type: List VisualEditor.DataRow
  List of picklist rows.

DynamicPickListRows()

Creates an instance of the VisualEditor.DynamicPickListRows class. You can then add rows by using the class’s addRow or addAllRows methods.

Signature

public DynamicPickListRows()
containsAllRows()  
Returns a Boolean value indicating whether all values of the picklist are included when a user does a type-ahead search query (true) 
or only those values initially displayed when the list is clicked on (false).

get(i)  
Returns a picklist element stored at the specified index.

dataGetRows()  
Returns a list of picklist items.

setDataRows(containsAllRows)  
Sets the value indicating whether all values of the picklist are included when a user does a type-ahead search query (true) or only 
those values initially displayed when the list is clicked on (false).

size()  
Returns the size of the list of VisualEditor.DynamicPickListRows.

sort()  
Sorts the list of VisualEditor.DynamicPickListRows.

**addAllRows (rows)**

Adds a list of picklist items to a dynamic picklist rendered in a Lightning component on a Lightning page.

**Signature**

```
public void addAllRows(List<VisualEditor.DataRow> rows)
```

**Parameters**

rows
  Type: List VisualEditor.DataRow
  List of picklist items.

**Return Value**

Type: void

**addRow (row)**

Adds a single picklist item to a dynamic picklist rendered in a Lightning component on a Lightning page.

**Signature**

```
public void addRow(VisualEditor.DataRow row)
```

**Parameters**

row
  Type: VisualEditor.DataRow
  A single picklist item.
Return Value
Type: void

clonel()

Signature
public Object clone()

Return Value
Type: Object

containsAllRows()
Returns a Boolean value indicating whether all values of the picklist are included when a user does a type-ahead search query (true) or only those values initially displayed when the list is clicked on (false).

Signature
public Boolean containsAllRows()

Return Value
Type: Boolean
A picklist in a Lightning component can display only the first 200 values of a list. If `containsAllRows` is set to false, when a user does a type-ahead search to find values in the picklist, the search will only look at those first 200 values that were displayed, not the complete set of picklist values.

geti()
Returns a picklist element stored at the specified index.

Signature
public VisualEditor.DataRow get(Integer i)

Parameters
i
  Type: Integer
  The index.

Return Value
Type: VisualEditor.DataRow
getDataRows ()
Returns a list of picklist items.

Signature
public List<VisualEditor.DataRow> getDataRows ()

Return Value
Type: List VisualEditor.DataRow

setContainsAllRows (containsAllRows)
Sets the value indicating whether all values of the picklist are included when a user does a type-ahead search query (true) or only those values initially displayed when the list is clicked on (false).

Signature
public void setContainsAllRows (Boolean containsAllRows)

Parameters
containsAllRows
Type: Boolean
Indicates if all values of the picklist are included in a type-ahead search query (true) or only those values initially displayed when the list is clicked on (false).

A picklist in a Lightning component can display only the first 200 values of a list. If containsAllRows is set to false, when a user does a type-ahead search to find values in the picklist, the search will only look at those first 200 values that were displayed, not the complete set of picklist values.

Return Value
Type: void

size ()
Returns the size of the list of VisualEditor.DynamicPickListRows.

Signature
public Integer size ()

Return Value
Type: Integer

sort ()
Sorts the list of VisualEditor.DynamicPickListRows.
Signature

```java
public void sort()
```

Return Value

Type: void

**Wave Namespace**

The classes in the Wave namespace are part of the CRM Analytics Analytics SDK, designed to facilitate querying CRM Analytics data from Apex code.

The following are the classes in the Wave namespace.

IN THIS SECTION:

- **QueryBuilder Class**
  The QueryBuilder class provides methods for constructing well-formed SAQL queries to pass to CRM Analytics.

- **QueryNode Class**
  Define each node of the query - such as projection, groups, order, filters. Execute the query.

- **ProjectionNode Class**
  Add aggregate functions to the query, or define an alias.

- **Templates Class**
  The Templates class provides methods for retrieving CRM Analytics template collections, individual templates, and template configurations.

- **TemplatesSearchOptions Class**
  The TemplatesSearchOptions class provides optional properties to filter the template collection.

**QueryBuilder Class**

The QueryBuilder class provides methods for constructing well-formed SAQL queries to pass to CRM Analytics.

**Namespace**

wave

**Usage**

Use QueryBuilder and its associated classes, Wave.ProjectionNode and Wave.QueryNode, to incrementally build your SAQL statement. For example:

```java
public static void executeApexQuery(String name){
    Wave.ProjectionNode[] projs = new Wave.ProjectionNode[]{
        Wave.QueryBuilder.get('State').alias('State'),
        Wave.QueryBuilder.get('City').alias('City'),
        Wave.QueryBuilder.get('Revenue').avg().alias('avg_Revenue'),
        Wave.QueryBuilder.get('Revenue').sum().alias('sum_Revenue'),
```
Examples

QueryBuilder is the core of this first phase of the CRM Analytics Apex SDK, so let’s take a closer look. Here’s a simple count query.

```
Wave.ProjectionNode[] projs = new
Wave.ProjectionNode[]{Wave.QueryBuilder.count().alias('c')};
String query = Wave.QueryBuilder.load('datasetId',
'datasetVersionId').group().foreach(projs).build('q');
```

The resulting SAQL query looks like this:

```
q = load "datasetId/datasetVersionId";
q = group q by all;
q = foreach q generate count as c;
```

Here’s a more complex example that uses a union statement.

```
Wave.ProjectionNode[] projs = new Wave.ProjectionNode[]{
Wave.QueryBuilder.get('Name'),
Wave.QueryBuilder.get('AnnualRevenue').alias('Revenue')};
Wave.QueryNode nodeOne =
Wave.QueryBuilder.load('datasetOne','datasetVersionOne').foreach(projs);
Wave.QueryNode nodeTwo = Wave.QueryBuilder.load('datasetTwo',
'datasetVersionTwo').foreach(projs);
String query = Wave.QueryBuilder.union(new List<Wave.QueryNode>{nodeOne,
nodeTwo}).build('q');
```

The resulting SAQL query has two projection streams, qa and qb.

```
qa = load "datasetOne/datasetVersionOne";
qa = foreach q generate Name,AnnualRevenue as Revenue;
qb = load "datasetTwo/datasetVersionTwo";
qb = foreach q generate Name,AnnualRevenue as Revenue;
q = union qa, qb;
```

IN THIS SECTION:

QueryBuilder Methods

QueryBuilder Methods

The following are methods for QueryBuilder.
IN THIS SECTION:

load(datasetID, datasetVersionID)
Load a stream from a dataset.

count()
Calculate the number of rows that match the query criteria.

get(projection)
Query by selecting specific attributes.

union(unionNodes)
Combine multiple result sets into one result set.

cogroup(cogroupNodes, groups)
Cogrouping means that two input streams are grouped independently and arranged side by side. Only data that exists in both groups appears in the results.

load(datasetID, datasetVersionID)
Load a stream from a dataset.

**Signature**

```java
public static wave.QueryNode load(String datasetID, String datasetVersionID)
```

**Parameters**

- `datasetID`
  - Type: `String`
  - The ID of the dataset.

- `datasetVersionID`
  - Type: `String`
  - The ID identifying the version of the dataset.

**Return Value**

Type: `wave.QueryNode`

**count()**
Calculate the number of rows that match the query criteria.

**Signature**

```java
public static wave.ProjectionNode count()
```

**Return Value**

Type: `wave.ProjectionNode`
`get (projection)`
Query by selecting specific attributes.

**Signature**
```java
public static wave.ProjectionNode get(String proj)
```

**Parameters**
- `proj`
  - Type: `String`
  - The name of the column to query.

**Return Value**
- Type: `wave.ProjectionNode`

`union (unionNodes)`
Combine multiple result sets into one result set.

**Signature**
```java
global static Wave.QueryNode union(List<Wave.QueryNode> unionNodes)
```

**Parameters**
- `unionNodes`
  - Type: `List<wave.QueryNode>`
  - List of nodes to combine.

**Return Value**
- Type: `wave.QueryNode`

`cogroup (cogroupNodes, groups)`
Cogrouping means that two input streams are grouped independently and arranged side by side. Only data that exists in both groups appears in the results.

**Signature**
```java
global static Wave.QueryNode cogroup(List<Wave.QueryNode> cogroupNodes, List<List<String>> groups)
```

**Parameters**
- `cogroupNodes`
  - Type: `wave.QueryNode`
List of nodes to group.

groups
Type: String
The type of grouping.

Return Value
Type: wave.QueryNode

QueryNode Class
Define each node of the query - such as projection, groups, order, filters. Execute the query.

Namespace
wave

Usage
Refer to the QueryBuilder example.

IN THIS SECTION:
QueryNode Methods

QueryNode Methods
The following are methods for QueryNode.

IN THIS SECTION:
build(streamName)
Build the query string represented by this QueryNode and assign it to a stream name.

foreach(projections)
Applies a set of expressions to every row in a dataset. This action is often referred to as projection.

group(groups)
Groups matched records (group by specific dataset attributes).

group()
Groups matched records (group by all).

order(orders)
Sorts in ascending or descending order on one or more fields.

cap(cap)
Limits the number of results that are returned.

filter(filterCondition)
Selects rows from a dataset based on a filter condition (a predicate).
filter(filterConditions)
Selects rows from a dataset based on multiple filter conditions (predicates).

execute(streamName)
Execute the query and return rows as JSON.

**build(streamName)**
Build the query string represented by this QueryNode and assign it to a stream name.

**Signature**

```
public String build(String streamName)
```

**Parameters**

- **streamName**
  - Type: String
  - The identifier for the stream - for example, "q".

**Return Value**

- Type: String
  - The SAQL query string represented by the QueryNode.

**foreach(projections)**
Applies a set of expressions to every row in a dataset. This action is often referred to as projection.

**Signature**

```
public wave.QueryNode foreach(List<wave.ProjectionNode> projections)
```

**Parameters**

- **projections**
  - Type: List<wave.ProjectionNode>
  - A list of ProjectionNodes to be added to this QueryNode.

**Return Value**

- Type: wave.QueryNode

**group(groups)**
Groups matched records (group by specific dataset attributes).

**Signature**

```
public wave.QueryNode group(List<String> groups)
```
### Parameters

**groups**

Type: `List<String>`

A list of expressions.

### Return Value

Type: `wave.QueryNode`

### Example

```java
Wave.ProjectionNode[] projs = new Wave.ProjectionNode[]{
    Wave.QueryBuilder.get('Name'),
    Wave.QueryBuilder.get('Revenue').sum().alias('REVENUE_SUM')};
ConnectApi.LiteralJson result = Wave.QueryBuilder.load('datasetId',
    'datasetVersionId').group(new String[]{'Name'}).foreach(projs).build('q');
```

### group()

Groups matched records (group by all).

**Signature**

```java
public wave.QueryNode group()
```

### Return Value

Type: `wave.QueryNode`

**Example**

```java
String query = Wave.QueryBuilder.load('datasetId',
    'datasetVersionId').group().foreach(projs).build('q');
```

### order(orders)

Sorts in ascending or descending order on one or more fields.

**Signature**

```java
public wave.QueryNode group(List<String> groups)
```

**Parameters**

**groups**

Type: `List<String>`

A list of column names and associated ascending or descending keywords, for example

```java
List<List<String>>{{'Name', 'asc'}, {'Revenue', 'desc'}}
```
Return Value
Type: `wave.QueryNode`

`cap(cap)`
Limits the number of results that are returned.

Signature
`global Wave.QueryNode cap(Integer cap)`

Parameters
`cap`
Type: `Integer`
The maximum number of rows to return.

Return Value
Type: `wave.QueryNode`

`filter(filterCondition)`
Selects rows from a dataset based on a filter condition (a predicate).

Signature
`public Wave.QueryNode filter(String filterCondition)`

Parameters
`filterCondition`
Type: `String`
For example: `filter('Name != 'My Name')`

Return Value
Type: `wave.QueryNode`

`filter(filterConditions)`
Selects rows from a dataset based on multiple filter conditions (predicates).

Signature
`public Wave.QueryNode filter(List<String> filterCondition)`
Parameters

*filterCondition*
Type: List<String>
A list of filter conditions.

Return Value
Type: wave.QueryNode

**execute(streamName)**
Execute the query and return rows as JSON.

Signature

```java
public ConnectApi.LiteralJson execute(String streamName)
```

Parameters

*streamName*
Type: String
The query stream to execute. For example:

```java
ConnectApi.LiteralJson result = Wave.QueryBuilder.load('datasetId', 'datasetVersionId').group().foreach(projs).execute('q');
```

Return Value
Type: ConnectApi.LiteralJson

**ProjectionNode Class**
Add aggregate functions to the query, or define an alias.

**Namespace**
wave on page 3643

**Usage**
Refer to the QueryBuilder example.

IN THIS SECTION:

ProjectionNode Methods

**ProjectionNode Methods**
The following are methods for ProjectionNode.
IN THIS SECTION:

- **sum()**
  Returns the sum of a numeric field.

- **avg()**
  Returns the average value of a numeric field.

- **min()**
  Returns the minimum value of a field.

- **max()**
  Returns the maximum value of a field.

- **count()**
  Returns the number of rows that match the query criteria.

- **unique()**
  Returns the count of unique values.

- **alias(name)**
  Define output column names.

**sum()**

Returns the sum of a numeric field.

**Signature**

```java
public wave.ProjectionNode sum()
```

**Return Value**

Type: `wave.ProjectionNode`

**avg()**

Returns the average value of a numeric field.

**Signature**

```java
public wave.ProjectionNode avg()
```

**Return Value**

Type: `wave.ProjectionNode`

**min()**

Returns the minimum value of a field.

**Signature**

```java
public wave.ProjectionNode min()
```
Return Value
Type: `wave.ProjectionNode`

```java
public wave.ProjectionNode max()
```

**max()**
Returns the maximum value of a field.

**Signature**
```
public wave.ProjectionNode max()
```

Return Value
Type: `wave.ProjectionNode`

```java
public wave.ProjectionNode count()
```

**count()**
Returns the number of rows that match the query criteria.

**Signature**
```
public wave.ProjectionNode count()
```

Return Value
Type: `wave.ProjectionNode`

```java
public wave.ProjectionNode unique()
```

**unique()**
Returns the count of unique values.

**Signature**
```
public wave.ProjectionNode unique()
```

Return Value
Type: `wave.ProjectionNode`

```java
public wave.ProjectionNode alias(String name)
```

**alias(name)**
Define output column names.

**Signature**
```
public wave.ProjectionNode alias(String name)
```
Parameters

name

Type: String

The name to use for this column. For example, this code defines the alias \( c \):

```java
```

Return Value

Type: \texttt{wave.ProjectionNode}

Templates Class

The Templates class provides methods for retrieving CRM Analytics template collections, individual templates, and template configurations.

Namespace

\texttt{Wave}

Usage

Use Templates and its associated class \texttt{Wave.TemplatesSearchOptions} to get CRM Analytics template information.

Examples

This code sample declares a method that returns a list of the template names.

```java
@AuraEnabled(cacheable=true)
public static void List<String> getTemplateNames() {
    Map<String, Object> o = Wave.Templates.getTemplates(new Wave.TemplatesSearchOptions());

    List<Object> templates = (List<Object>) o.get('templates');
    List<String> names = new List<String>();
    for (Object templateObj : templates) {
        names.add((String) ((Map<String, Object>) templateObj.get('name')));
    }
    return names;
}
```

Adding the \texttt{@AuraEnabled} annotation allows Lightning Web Components to access Templates methods directly.

For example, in the lwc.js file:

```javascript
import getTemplates from '@salesforce/apex/Wave.Templates.getTemplates';
export default class Templates extends LightningElement {
    @wire(getTemplates, {
        // specifying 'options' is optional
        options: {
            // values in TemplatesSearchOptions go here; all optional
            type: 'app'
        }
    }
```
Templates Methods

The following are methods for Templates.

**getTemplate(templateIdOrApiName)**

Gets a CRM Analytics template by the specified ID or API name. The returned template is a map of the template JSON attributes as name/value pairs.

**getTemplateConfig(templateIdOrApiName)**

Gets the CRM Analytics template configuration by the specified ID or API name. The returned template configuration is a map of the JSON attributes as name/value pairs.

**getTemplates(options)**

Get a filtered collection of CRM Analytics templates using search options.

**getTemplates()**

Gets all CRM Analytics templates.

---

**getTemplate(templateIdOrApiName)**

Gets a CRM Analytics template by the specified ID or API name. The returned template is a map of the template JSON attributes as name/value pairs.

**Signature**

```java
public static Map<String, Object> getTemplate(String templateIdOrApiName)
```

**Parameters**

- **templateIdOrApiName**
  - Type: `String`
  - The template ID or API name of the template to retrieve.

**Return Value**

- Type: `Map<String, Object>`
A map of the template JSON attribute name/value pairs, where the name is a string with an object value. For attributes details, see TemplateRepresentation.

Example

```java
String templateName = (String) Wave.Templates.getTemplate(templateId).get('name');
```

**getTemplateConfig(templateIdOrApiName)**

Gets the CRM Analytics template configuration by the specified ID or API name. The returned template configuration is a map of the JSON attributes as name/value pairs.

**Signature**

```java
public static Map<String, Object> getTemplateConfig(String templateIdOrApiName)
```

**Parameters**

- `templateIdOrApiName`  
  Type: `String`  
  The template ID or developer name to retrieve the template configuration for.

**Return Value**

- Type: `Map<String, Object>`  
  A map of template configuration JSON attribute names and the object values. For attribute details, see TemplateConfigurationRepresentation.

Example

```java
Map<String, Object> templateVariables = (Map<String, Object>) Wave.Templates.getTemplateConfig(templateId).get('variables');
```

**getTemplates(options)**

Get a filtered collection of CRM Analytics templates using search options.

**Signature**

```java
public static Map<String, Object> getTemplates(Wave.TemplatesSearchOptions options)
```

**Parameters**

- `options`  
  Type: `Wave.TemplatesSearchOptions` on page 3657  
  The search options to use for filtering the template collection.
Return Value
Type: Map<String,Object>
A map of template names and the template object values. For template collection details, see TemplateCollectionRepresentation.

Example
Map<String, Object> templatesMap = Wave.Templates.getTemplates(new Wave.TemplatesSearchOptions());

getTemplates()
Gets all CRM Analytics templates.

Signature
public static Map<String, Object> getTemplates()

Return Value
Type: Map<String,Object>
A map of template names and the template object values. For template collection details, see TemplateCollectionRepresentation.

Example
Map<String, Object> templatesMap = Wave.Templates.getTemplates();

TemplatesSearchOptions Class
The TemplatesSearchOptions class provides optional properties to filter the template collection.

Namespace
Wave

Usage
Use TemplatesSearchOptions with Wave.Templates class to filter the CRM Analytics template collection returned. For example:

```java
public static void List<String> getAppTemplates() {
    Wave.TemplatesSearchOptions tsOptions = new Wave.TemplatesSearchOptions();
    tsOptions.type = 'app';

    Map<String, Object> o = Wave.Templates.getTemplates(tsOptions);
    List<Object> appTemplates = (List<Object>) o.get('templates');
    List<String> names = new List<String>();
    for (Object templateObj : appTemplates) {
        names.add((String) ((Map<String, Object>) templateObj).get('name'));
    }
}
TemplatesSearchOptions Properties

The following are properties for TemplatesSearchOptions.

filterGroup
Specifies the Connect API filter group for CRM Analytics template search options.

options
Specifies the template visibility option to filter the CRM Analytics template collection by.

type
Sets the template type to filter the CRM Analytics template collection by.

filterGroup

Specifies the Connect API filter group for CRM Analytics template search options.

Signature

public String filterGroup {get; set;}

Property Value

Type: String

Uses the ConnectFilterGroupEnum values.

Example

```
Wave.TemplateSearchOptions tsOptions = new Wave.TemplateSearchOptions();
nullsOptions.filterGroup = 'small';
```

options

Specifies the template visibility option to filter the CRM Analytics template collection by.

Signature

public String options {get; set;}

return names;
}
Property Value
Type: String
Uses the ConnectWaveTemplateVisibilityOptionsEnum values. Valid values are CreateApp, ViewOnly, and ManageableOnly.

Example
Wave.TemplateSearchOptions tsOptions = new Wave.TemplateSearchOptions();
tsOptions.options = 'ViewOnly';

type
Sets the template type to filter the CRM Analytics template collection by.

Signature
public String type {get; set;}

Property Value
Type: String
Uses the ConnectWaveTemplateTypeEnum values. Valid values are app, dashboard, embedded, and lens.

Example
Wave.TemplateSearchOptions tsOptions = new Wave.TemplateSearchOptions();
tsOptions.type = 'app';

Appendices

IN THIS SECTION:
Shipping Invoice Example
Reserved Keywords
These words can be used only as keywords.
Documentation Typographical Conventions
Glossary

Shipping Invoice Example
This appendix provides an example of an Apex application. This is a more complex example than the Hello World example.

• Shipping Invoice Walk-Through
• Shipping Invoice Example Code
Shipping Invoice Example Walk-Through

The sample application in this section includes traditional Salesforce functionality blended with Apex. Many of the syntactic and semantic features of Apex, along with common idioms, are illustrated in this application.

**Note:** The Shipping Invoice sample requires custom objects. You can either create these on your own, or download the objects and Apex code as an unmanaged package from the Salesforce AppExchange. To obtain the sample assets in your org, install the Apex Tutorials Package. This package also contains sample code and objects for the Apex Quick Start.

**Scenario**

In this sample application, the user creates a new shipping invoice, or order, and then adds items to the invoice. The total amount for the order, including shipping cost, is automatically calculated and updated based on the items added or deleted from the invoice.

**Data and Code Models**

This sample application uses two new objects: Item and Shipping_invoice.

The following assumptions are made:

- Item A cannot be in both orders shipping_invoice1 and shipping_invoice2. Two customers cannot obtain the same (physical) product.
- The tax rate is 9.25%.
- The shipping rate is 75 cents per pound.
- Once an order is over $100, the shipping discount is applied (shipping becomes free).

The fields in the Item custom object include:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>String</td>
<td>The name of the item</td>
</tr>
<tr>
<td>Price</td>
<td>Currency</td>
<td>The price of the item</td>
</tr>
<tr>
<td>Quantity</td>
<td>Number</td>
<td>The number of items in the order</td>
</tr>
<tr>
<td>Weight</td>
<td>Number</td>
<td>The weight of the item, used to calculate shipping costs</td>
</tr>
<tr>
<td>Shipping_invoice</td>
<td>Master-Detail (shipping_invoice)</td>
<td>The order this item is associated with</td>
</tr>
</tbody>
</table>

The fields in the Shipping_invoice custom object include:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>String</td>
<td>The name of the shipping invoice/order</td>
</tr>
<tr>
<td>Subtotal</td>
<td>Currency</td>
<td>The subtotal</td>
</tr>
</tbody>
</table>
### Table of Key Terms

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GrandTotal</td>
<td>Currency</td>
<td>The total amount, including tax and shipping</td>
</tr>
<tr>
<td>Shipping</td>
<td>Currency</td>
<td>The amount charged for shipping (assumes $0.75 per pound)</td>
</tr>
<tr>
<td>ShippingDiscount</td>
<td>Currency</td>
<td>Only applied once when subtotal amount reaches $100</td>
</tr>
<tr>
<td>Tax</td>
<td>Currency</td>
<td>The amount of tax (assumes 9.25%)</td>
</tr>
<tr>
<td>TotalWeight</td>
<td>Number</td>
<td>The total weight of all items</td>
</tr>
</tbody>
</table>

All of the Apex for this application is contained in triggers. This application has the following triggers:

<table>
<thead>
<tr>
<th>Object</th>
<th>Trigger Name</th>
<th>When Runs</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item</td>
<td>Calculate</td>
<td>after insert, after update, after delete</td>
<td>Updates the shipping invoice, calculates the totals and shipping</td>
</tr>
<tr>
<td>Shipping_invoice</td>
<td>ShippingDiscount</td>
<td>after update</td>
<td>Updates the shipping invoice, calculating if there is a shipping discount</td>
</tr>
</tbody>
</table>

The following is the general flow of user actions and when triggers run:

**Flow of user action and triggers for the shopping cart application**

1. User clicks **Orders > New**, names the shipping invoice and clicks **Save**.
2. User clicks **New Item**, fills out information, and clicks **Save**.

3. Calculate trigger runs. Part of the Calculate trigger updates the shipping invoice.

4. ShippingDiscount trigger runs.

5. User can then add, delete or change items in the invoice.

In **Shipping Invoice Example Code** both of the triggers and the test class are listed. The comments in the code explain the functionality.

### Testing the Shipping Invoice Application

Before an application can be included as part of a package, 75% of the code must be covered by unit tests. Therefore, one piece of the shipping invoice application is a class used for testing the triggers.

The test class verifies the following actions are completed successfully:

- Inserting items
- Updating items
- Deleting items
- Applying shipping discount
- Negative test for bad input

### Shipping Invoice Example Code

The following triggers and test class make up the shipping invoice example application:

- Calculate trigger
- ShippingDiscount trigger
- Test class

#### Calculate Trigger

```
trigger calculate on Item__c (after insert, after update, after delete) {

// Use a map because it doesn't allow duplicate values
Map<ID, Shipping_Invoice__C> updateMap = new Map<ID, Shipping_Invoice__C>();

// Set this integer to -1 if we are deleting
Integer subtract;

// Populate the list of items based on trigger type
List<Item__c> itemList;
if(trigger.isInsert || trigger.isUpdate) {
    itemList = Trigger.new;
    subtract = 1;
} else if(trigger.isDelete) {
    // Note -- there is no trigger.new in delete
    itemList = trigger.old;
    subtract = -1;
}
```

// Access all the information we need in a single query
// rather than querying when we need it.
// This is a best practice for bulkifying requests

set<Id> AllItems = new set<id>();

for(item__c i :itemList){
    // Assert numbers are not negative.
    // None of the fields would make sense with a negative value
    System.assert(i.quantity__c > 0, 'Quantity must be positive');
    System.assert(i.weight__c >= 0, 'Weight must be non-negative');
    System.assert(i.price__c >= 0, 'Price must be non-negative');

    // If there is a duplicate Id, it won't get added to a set
    AllItems.add(i.Shipping_Invoice__C);
}

// Accessing all shipping invoices associated with the items in the trigger
List<Shipping_Invoice__c> AllShippingInvoices = [SELECT Id, ShippingDiscount__c,
                                                 SubTotal__c, TotalWeight__c, Tax__c, GrandTotal__c
                                                 FROM Shipping_Invoice__c WHERE Id IN :AllItems];

// Take the list we just populated and put it into a Map.
// This will make it easier to look up a shipping invoice
// because you must iterate a list, but you can use lookup for a map,
Map<ID, Shipping_Invoice__c> SIMap = new Map<ID, Shipping_Invoice__c>();

for(Shipping_Invoice__c sc : AllShippingInvoices)
{
    SIMap.put(sc.id, sc);
}

// Process the list of items
if(Trigger.isUpdate)
{
    // Treat updates like a removal of the old item and addition of the
    // revised item rather than figuring out the differences of each field
    // and acting accordingly.
    // Note updates have both trigger.new and trigger.old
    for(Integer x = 0; x < Trigger.old.size(); x++)
    {
        Shipping_Invoice__c myOrder;
        myOrder = SIMap.get(trigger.old[x].Shipping_Invoice__C);

        // Decrement the previous value from the subtotal and weight.
        myOrder.SubTotal__c -= (trigger.old[x].price__c * trigger.old[x].quantity__c);
        myOrder.TotalWeight__c -= (trigger.old[x].weight__c * trigger.old[x].quantity__c);

        // Increment the new subtotal and weight.
        myOrder.SubTotal__c += (trigger.new[x].price__c * trigger.new[x].quantity__c);
    }
}
```apex
trigger.new[x].quantity__c);
myOrder.TotalWeight__c += (trigger.new[x].weight__c *
    trigger.new[x].quantity__c);
}

for(Shipping_Invoice__C myOrder : AllShippingInvoices)
{

    // Set tax rate to 9.25% Please note, this is a simple example.
    // Generally, you would never hard code values.
    // Leveraging Custom Settings for tax rates is a best practice.
    // See Custom Settings in the Apex Developer Guide
    // for more information.
    myOrder.Tax__c = myOrder.Subtotal__c * .0925;

    // Reset the shipping discount
    myOrder.ShippingDiscount__c = 0;

    // Set shipping rate to 75 cents per pound.
    // Generally, you would never hard code values.
    // Leveraging Custom Settings for the shipping rate is a best practice.
    // See Custom Settings in the Apex Developer Guide
    // for more information.
    myOrder.Shipping__c = (myOrder.totalWeight__c * .75);
    myOrder.GrandTotal__c = myOrder.SubTotal__c + myOrder.tax__c +
        myOrder.Shipping__c;

    updateMap.put(myOrder.id, myOrder);
}
else
{
    for(Item__c itemToProcess : itemList)
    {
        Shipping_Invoice__C myOrder;

        // Look up the correct shipping invoice from the ones we got earlier
        myOrder = SIMap.get(itemToProcess.Shipping_Invoice__C);
        myOrder.SubTotal__c += (itemToProcess.price__c *
            itemToProcess.quantity__c * subtract);
        myOrder.TotalWeight__c += (itemToProcess.weight__c *
            itemToProcess.quantity__c * subtract);
    }

    for(Shipping_Invoice__C myOrder : AllShippingInvoices)
    {

        // Set tax rate to 9.25% Please note, this is a simple example.
        // Generally, you would never hard code values.
        // Leveraging Custom Settings for tax rates is a best practice.
        // See Custom Settings in the Apex Developer Guide
        // for more information.
        myOrder.Tax__c = myOrder.Subtotal__c * .0925;

        // Reset shipping discount
```
myOrder.ShippingDiscount__c = 0;

// Set shipping rate to 75 cents per pound.
// Generally, you would never hard code values.
// Leveraging Custom Settings for the shipping rate is a best practice.
// See Custom Settings in the Apex Developer Guide for more information.
myOrder.Shipping__c = (myOrder.totalWeight__c * .75);
myOrder.GrandTotal__c = myOrder.SubTotal__c + myOrder.tax__c +
    myOrder.Shipping__c;

updateMap.put(myOrder.id, myOrder);

// Only use one DML update at the end.
// This minimizes the number of DML requests generated from this trigger.
update updateMap.values();

ShippingDiscount Trigger

trigger ShippingDiscount on Shipping_Invoice__C (before update) {
    // Free shipping on all orders greater than $100
    for(Shipping_Invoice__C myShippingInvoice : Trigger.new)
    {
        if((myShippingInvoice.subtotal__c >= 100.00) &&
           (myShippingInvoice.ShippingDiscount__c == 0))
        {
            myShippingInvoice.ShippingDiscount__c = myShippingInvoice.Shipping__c * -1;
            myShippingInvoice.GrandTotal__c += myShippingInvoice.ShippingDiscount__c;
        }
    }
}

Shipping Invoice Test

@IsTest
private class TestShippingInvoice{

    // Test for inserting three items at once
    public static testmethod void testBulkItemInsert(){
        // Create the shipping invoice. It's a best practice to either use defaults
        // or to explicitly set all values to zero so as to avoid having
        // extraneous data in your test.
        Shipping_Invoice__C order1 = new Shipping_Invoice__C(subtotal__c = 0,
            totalweight__c = 0, grandtotal__c = 0,
            ShippingDiscount__c = 0, Shipping__c = 0, tax__c = 0);

        // Insert the order and populate with items
insert Order1;
List<Item__c> list1 = new List<Item__c>();
Item__c item1 = new Item__c(Price__c = 10, weight__c = 1, quantity__c = 1,
Shipping_Invoice__C = order1.id);
Item__c item2 = new Item__c(Price__c = 25, weight__c = 2, quantity__c = 1,
Shipping_Invoice__C = order1.id);
Item__c item3 = new Item__c(Price__c = 40, weight__c = 3, quantity__c = 1,
Shipping_Invoice__C = order1.id);
list1.add(item1);
list1.add(item2);
list1.add(item3);
insert list1;

// Retrieve the order, then do assertions
order1 = [SELECT id, subtotal__c, tax__c, shipping__c, totalweight__c,
grandtotal__c, shippingdiscount__c
FROM Shipping_Invoice__C
WHERE id = :order1.id];
System.assert(order1.subtotal__c == 75,
'Order subtotal was not $75, but was ' + order1.subtotal__c);
System.assert(order1.tax__c == 6.9375,
'Order tax was not $6.9375, but was ' + order1.tax__c);
System.assert(order1.shipping__c == 4.50,
'Order shipping was not $4.50, but was ' + order1.shipping__c);
System.assert(order1.totalweight__c == 6.00,
'Order weight was not 6 but was ' + order1.totalweight__c);
System.assert(order1.grandtotal__c == 86.4375,
'Order grand total was not $86.4375 but was ' + order1.grandtotal__c);
System.assert(order1.shippingdiscount__c == 0,
'Order shipping discount was not $0 but was ' + order1.shippingdiscount__c);

// Test for updating three items at once
public static testmethod void testBulkItemUpdate(){

// Create the shipping invoice. It's a best practice to either use defaults
// or to explicitly set all values to zero so as to avoid having
// extraneous data in your test.
Shipping_Invoice__C order1 = new Shipping_Invoice__C(subtotal__c = 0,
totalweight__c = 0, grandtotal__c = 0,
ShippingDiscount__c = 0, Shipping__c = 0, tax__c = 0);

// Insert the order and populate with items.
insert Order1;
List<Item__c> list1 = new List<Item__c>();
Item__c item1 = new Item__c(Price__c = 1, weight__c = 1, quantity__c = 1,
Shipping_Invoice__C = order1.id);
Item__c item2 = new Item__c(Price__c = 2, weight__c = 2, quantity__c = 1,
Shipping_Invoice__C = order1.id);
Item__c item3 = new Item__c(Price__c = 4, weight__c = 3, quantity__c = 1,
Shipping_Invoice__C = order1.id);
list1.add(item1);
list1.add(item2);
list1.add(item3);
insert list1;

// Update the prices on the 3 items
list1[0].price__c = 10;
list1[1].price__c = 25;
list1[2].price__c = 40;
update list1;

// Access the order and assert items updated
order1 = [SELECT id, subtotal__c, tax__c, shipping__c, totalweight__c,
          grandtotal__c, shippingdiscount__c
          FROM Shipping_Invoice__C
          WHERE Id = :order1.Id];

System.assert(order1.subtotal__c == 75,
              'Order subtotal was not $75, but was ' + order1.subtotal__c);
System.assert(order1.tax__c == 6.9375,
              'Order tax was not $6.9375, but was ' + order1.tax__c);
System.assert(order1.shipping__c == 4.50,
              'Order shipping was not $4.50, but was ' + order1.shipping__c);
System.assert(order1.totalweight__c == 6.00,
              'Order weight was not 6 but was ' + order1.totalweight__c);
System.assert(order1.grandtotal__c == 86.4375,
              'Order grand total was not $86.4375 but was ' + order1.grandtotal__c);
System.assert(order1.shippingdiscount__c == 0,
              'Order shipping discount was not $0 but was ' + order1.shippingdiscount__c);

}

// Test for deleting items
public static testmethod void testBulkItemDelete(){

    // Create the shipping invoice. It's a best practice to either use defaults
    // or to explicitly set all values to zero so as to avoid having
    // extraneous data in your test.
    Shipping_Invoice__C order1 = new Shipping_Invoice__C(subtotal__c = 0,
                                                        totalweight__c = 0,
                                                        grandtotal__c = 0,
                                                        ShippingDiscount__c = 0,
                                                        Shipping__c = 0,
                                                        tax__c = 0);

    // Insert the order and populate with items
    insert Order1;
    List<Item__c> list1 = new List<Item__c>();
    Item__c item1 = new Item__c(Price__c = 10, weight__c = 1, quantity__c = 1,
                                Shipping_Invoice__C = order1.id);
    Item__c item2 = new Item__c(Price__c = 25, weight__c = 2, quantity__c = 1,
                                Shipping_Invoice__C = order1.id);
    Item__c item3 = new Item__c(Price__c = 40, weight__c = 3, quantity__c = 1,
                                Shipping_Invoice__C = order1.id);
Item__c itemA = new Item__C(Price__c = 1, weight__c = 3, quantity__c = 1,
Shipping_Invoice__C = order1.id);
Item__c itemB = new Item__C(Price__c = 1, weight__c = 3, quantity__c = 1,
Shipping_Invoice__C = order1.id);
Item__c itemC = new Item__C(Price__c = 1, weight__c = 3, quantity__c = 1,
Shipping_Invoice__C = order1.id);
Item__c itemD = new Item__C(Price__c = 1, weight__c = 3, quantity__c = 1,
Shipping_Invoice__C = order1.id);
list1.add(item1);
list1.add(item2);
list1.add(item3);
list1.add(itemA);
list1.add(itemB);
list1.add(itemC);
list1.add(itemD);
insert list1;

// Seven items are now in the shipping invoice.
// The following deletes four of them.
List<Item__c> list2 = new List<Item__c>();
list2.add(itemA);
list2.add(itemB);
list2.add(itemC);
list2.add(itemD);
delete list2;

// Retrieve the order and verify the deletion
order1 = [SELECT id, subtotal__c, tax__c, shipping__c, totalweight__c,
grandtotal__c, shippingdiscount__c
FROM Shipping_Invoice__C
WHERE Id = :order1.Id];
System.assert(order1.subtotal__c == 75,
'Order subtotal was not $75, but was ' + order1.subtotal__c);
System.assert(order1.tax__c == 6.9375,
'Order tax was not $6.9375, but was ' + order1.tax__c);
System.assert(order1.shipping__c == 4.50,
'Order shipping was not $4.50, but was ' + order1.shipping__c);
System.assert(order1.totalweight__c == 6.00,
'Order weight was not 6 but was ' + order1.totalweight__c);
System.assert(order1.grandtotal__c == 86.4375,
'Order grand total was not $86.4375 but was '
+ order1.grandtotal__c);
System.assert(order1.shippingdiscount__c == 0,
'Order shipping discount was not $0 but was '
+ order1.shippingdiscount__c);
}

// Testing free shipping
public static testmethod void testFreeShipping(){

  // Create the shipping invoice. It's a best practice to either use defaults
  // or to explicitly set all values to zero so as to avoid having
  // extraneous data in your test.
  Shipping_Invoice__C order1 = new Shipping_Invoice__C(subtotal__c = 0,
totalweight__c = 0, grandtotal__c = 0,
ShippingDiscount__c = 0, Shipping__c = 0, tax__c = 0);

// Insert the order and populate with items.
insert Order1;
List<Item__c> list1 = new List<Item__c>();
Item__c item1 = new Item__c(Price__c = 10, weight__c = 1,
    quantity__c = 1, Shipping_Invoice__C = order1.id);
Item__c item2 = new Item__c(Price__c = 25, weight__c = 2,
    quantity__c = 1, Shipping_Invoice__C = order1.id);
Item__c item3 = new Item__c(Price__c = 40, weight__c = 3,
    quantity__c = 1, Shipping_Invoice__C = order1.id);
list1.add(item1);
list1.add(item2);
list1.add(item3);
insert list1;

// Retrieve the order and verify free shipping not applicable
order1 = [SELECT id, subtotal__c, tax__c, shipping__c, totalweight__c,
            grandtotal__c, shippingdiscount__c
        FROM Shipping_Invoice__C
        WHERE Id = :order1.Id];

// Free shipping not available on $75 orders
System.assert(order1.subtotal__c == 75,
    'Order subtotal was not $75, but was ' + order1.subtotal__c);
System.assert(order1.tax__c == 6.9375,
    'Order tax was not $6.9375, but was ' + order1.tax__c);
System.assert(order1.shipping__c == 4.50,
    'Order shipping was not $4.50, but was ' + order1.shipping__c);
System.assert(order1.totalweight__c == 6.00,
    'Order weight was not 6 but was ' + order1.totalweight__c);
System.assert(order1.grandtotal__c == 86.4375,
    'Order grand total was not $86.4375 but was ' + order1.grandtotal__c);
System.assert(order1.shippingdiscount__c == 0,
    'Order shipping discount was not $0 but was ' + order1.shippingdiscount__c);

// Add items to increase subtotal
item1 = new Item__c(Price__c = 25, weight__c = 20, quantity__c = 1,
    Shipping_Invoice__C = order1.id);
insert item1;

// Retrieve the order and verify free shipping is applicable
order1 = [SELECT id, subtotal__c, tax__c, shipping__c, totalweight__c,
            grandtotal__c, shippingdiscount__c
        FROM Shipping_Invoice__C
        WHERE Id = :order1.Id];

// Order total is now at $100, so free shipping should be enabled
System.assert(order1.subtotal__c == 100,
    'Order subtotal was not $100, but was ' + order1.subtotal__c);
System.assert(order1.tax__c == 9.25,
'Order tax was not $9.25, but was ' + order1.tax__c);
System.assert(order1.shipping__c == 19.50,
'Order shipping was not $19.50, but was ' + order1.shipping__c);
System.assert(order1.totalweight__c == 26.00,
'Order weight was not 26 but was ' + order1.totalweight__c);
System.assert(order1.grandtotal__c == 109.25,
'Order grand total was not $86.4375 but was ' + order1.grandtotal__c);
System.assert(order1.shippingdiscount__c == -19.50,
'Order shipping discount was not -$19.50 but was ' + order1.shippingdiscount__c);
}

// Negative testing for inserting bad input
public static testmethod void testNegativeTests(){

    // Create the shipping invoice. It's a best practice to either use defaults
    // or to explicitly set all values to zero so as to avoid having
    // extraneous data in your test.
    Shipping_Invoice__C order1 = new Shipping_Invoice__C(subtotal__c = 0,
    totalweight__c = 0, grandtotal__c = 0,
    ShippingDiscount__c = 0, Shipping__c = 0, tax__c = 0);

    // Insert the order and populate with items.
    insert Order1;
    Item__c item1 = new Item__C(Price__c = -10, weight__c = 1, quantity__c = 1,
    Shipping_Invoice__C = order1.id);
    Item__c item2 = new Item__C(Price__c = 25, weight__c = -2, quantity__c = 1,
    Shipping_Invoice__C = order1.id);
    Item__c item3 = new Item__C(Price__c = 40, weight__c = 3, quantity__c = -1,
    Shipping_Invoice__C = order1.id);
    Item__c item4 = new Item__C(Price__c = 40, weight__c = 3, quantity__c = 0,
    Shipping_Invoice__C = order1.id);

    try{
        insert item1;
    }
    catch(Exception e)
    {
        system.assert(e.getMessage().contains('Price must be non-negative'),
        'Price was negative but was not caught');
    }

    try{
        insert item2;
    }
    catch(Exception e)
    {
        system.assert(e.getMessage().contains('Weight must be non-negative'),
        'Weight was negative but was not caught');
    }

    try{
Reserved Keywords

These words can be used only as keywords.

<table>
<thead>
<tr>
<th>Table 3: Reserved Keywords</th>
</tr>
</thead>
<tbody>
<tr>
<td>abstract</td>
</tr>
<tr>
<td>activate</td>
</tr>
<tr>
<td>and</td>
</tr>
<tr>
<td>any</td>
</tr>
<tr>
<td>array</td>
</tr>
<tr>
<td>as</td>
</tr>
<tr>
<td>asc</td>
</tr>
<tr>
<td>autonomous</td>
</tr>
<tr>
<td>begin</td>
</tr>
<tr>
<td>bigdecimal</td>
</tr>
<tr>
<td>blob</td>
</tr>
<tr>
<td>boolean</td>
</tr>
<tr>
<td>break</td>
</tr>
<tr>
<td>bulk</td>
</tr>
<tr>
<td>by</td>
</tr>
<tr>
<td>byte</td>
</tr>
<tr>
<td>case</td>
</tr>
<tr>
<td>cast</td>
</tr>
<tr>
<td>catch</td>
</tr>
</tbody>
</table>
These words are special types of keywords that aren’t reserved words and can be used as identifiers.

- after
- before
- count
- excludes
- first
- includes
- last
- order
- sharing
- with

Documentation Typographical Conventions

Apex and Visualforce documentation uses the following typographical conventions.
<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Courier font</td>
<td>In descriptions of syntax, monospace font indicates items that you should type as shown, except for brackets. For example:</td>
</tr>
<tr>
<td></td>
<td><strong>Public class</strong> HelloWorld</td>
</tr>
<tr>
<td>Italics</td>
<td>In descriptions of syntax, italics represent variables. You supply the actual value. In the following example, three values need to be supplied:</td>
</tr>
<tr>
<td></td>
<td><em>datatype</em> <em>variable_name</em> = <em>value</em>;</td>
</tr>
<tr>
<td></td>
<td>If the syntax is bold and italic, the text represents a code element that needs a value supplied by you, such as a class name or variable value:</td>
</tr>
<tr>
<td></td>
<td><strong>public static class</strong> YourClassHere { ... }</td>
</tr>
<tr>
<td>Bold Courier font</td>
<td>In code samples and syntax descriptions, bold courier font emphasizes a portion of the code or syntax.</td>
</tr>
<tr>
<td>&lt;&gt;</td>
<td>In descriptions of syntax, less-than and greater-than symbols (&lt; &gt;) are typed exactly as shown.</td>
</tr>
<tr>
<td></td>
<td><code>&lt;apex:pageBlockTable value=&quot; {!account.Contacts} &quot; var=&quot;contact&quot;&gt;</code></td>
</tr>
<tr>
<td></td>
<td><code>&lt;apex:column value=&quot; {!contact.Name}&quot;/&gt;</code></td>
</tr>
<tr>
<td></td>
<td><code>&lt;apex:column value=&quot; {!contact.MailingCity}&quot;/&gt;</code></td>
</tr>
<tr>
<td></td>
<td><code>&lt;apex:column value=&quot; {!contact.Phone}&quot;/&gt;</code></td>
</tr>
<tr>
<td></td>
<td><code>&lt;/apex:pageBlockTable&gt;</code></td>
</tr>
<tr>
<td>[]</td>
<td>In descriptions of syntax, braces ({ }) are typed exactly as shown.</td>
</tr>
<tr>
<td></td>
<td><code>&lt;apex:page&gt;</code></td>
</tr>
<tr>
<td></td>
<td>Hello {!$User.FirstName}!</td>
</tr>
<tr>
<td></td>
<td><code>&lt;/apex:page&gt;</code></td>
</tr>
<tr>
<td>[ ]</td>
<td>In descriptions of syntax, anything included in brackets is optional. In the following example, specifying <em>value</em> is optional:</td>
</tr>
<tr>
<td></td>
<td><em>data_type</em> <em>variable_name</em> = <em>value</em>;</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>In the following example, you can create a new unpopulated set in one of two ways, or you can populate the set:</td>
</tr>
<tr>
<td></td>
<td>Set&lt;data_type&gt; <em>set_name</em></td>
</tr>
<tr>
<td></td>
<td>[= new Set&lt;data_type&gt; ()]</td>
</tr>
<tr>
<td></td>
<td>[= new Set&lt;data_type&gt; {value, value2...}]</td>
</tr>
<tr>
<td></td>
<td>;</td>
</tr>
</tbody>
</table>

**Glossary**

A | B | C | D | E | F | G | H | I | J | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z

3673
Administrator (System Administrator)
One or more individuals in your organization who can configure and customize the application. Users assigned to the System Administrator profile have administrator privileges.

AJAX Toolkit
A JavaScript wrapper around the API that allows you to execute any API call and access any object you have permission to view from within JavaScript code. For more information, see the AJAX Toolkit Developer Guide.

Anti-Join
An anti-join is a subquery on another object in a `NOT IN` clause in a SOQL query. You can use anti-joins to create advanced queries. See also Semi-Join.

Anonymous Block, Apex
Apex code that doesn't get stored in Salesforce, but that can be compiled and executed by using the `ExecuteAnonymousResult()` API call, or the equivalent in the AJAX Toolkit.

Ant Migration Tool
A toolkit that allows you to write an Apache Ant build script for migrating Lightning Platform components between a local file system and a Salesforce organization.

Apex
Apex is a strongly typed, object-oriented programming language that allows developers to execute flow and transaction control statements on the Lightning Platform server in conjunction with calls to the Lightning Platform API. Using syntax that looks like Java and acts like database-stored procedures, Apex enables developers to add business logic to most system events, including button clicks, related record updates, and Visualforce pages. Apex code can be initiated by Web service requests and from triggers on objects.

Apex Connector Framework
The Apex Connector Framework is a set of classes and methods in the DataSource namespace for creating a custom adapter for Salesforce Connect. Create a custom adapter to connect to data that's stored outside your Salesforce org when the other available Salesforce Connect adapters aren't suitable for your needs.

Apex-Managed Sharing
Enables developers to programmatically manipulate sharing to support their application’s behavior. Apex-managed sharing is only available for custom objects.

Apex Page
See Visualforce page.

App
Short for “application.” A collection of components such as tabs, reports, dashboards, and Visualforce pages that address a specific business need. Salesforce provides standard apps such as Sales and Service. You can customize the standard apps to match the way you work. In addition, you can package an app and upload it to AppExchange along with related components such as custom fields, custom tabs, and custom objects. Then, you can make the app available to other Salesforce users from AppExchange.

AppExchange
The AppExchange is a sharing interface from Salesforce that allows you to browse and share apps and services for the Lightning Platform.

Application Programming Interface (API)
The interface that a computer system, library, or application provides to allow other computer programs to request services from it and exchange data.

Approval Process
An approval process automates how records are approved in Salesforce. An approval process specifies each step of approval, including who to request approval from and what to do at each point of the process.
Asynchronous Calls
A call that doesn’t return results immediately because the operation can take a long time. Calls in the Metadata API and Bulk API 2.0 are asynchronous.

B
Batch Apex
The ability to perform long, complex operations on many records at a scheduled time using Apex.

Beta, Managed Package
In the context of managed packages, a beta managed package is an early version of a managed package distributed to a sampling of your intended audience to test it.

Bulk API 2.0
The REST-based Bulk API 2.0 is optimized for processing large sets of data. It allows you to query, insert, update, upsert, or delete a large number of records asynchronously by submitting a job that is processed in the background by Salesforce. See also SOAP API.

C
Callout, Apex
An Apex callout enables you to tightly integrate your Apex with an external service by making a call to an external Web service or sending a HTTP request from Apex code and then receiving the response.

Child Relationship
A relationship that has been defined on an sObject that references another sObject as the “one” side of a one-to-many relationship. For example, contacts, opportunities, and tasks have child relationships with accounts.
See also sObject.

Class, Apex
A template or blueprint from which Apex objects are created. Classes consist of other classes, user-defined methods, variables, exception types, and static initialization code. In most cases, Apex classes are modeled on their counterparts in Java.

Client App
An app that runs outside the Salesforce user interface and uses only the Lightning Platform API or Bulk API 2.0. It typically runs on a desktop or mobile device. These apps treat the platform as a data source, using the development model of whatever tool and platform for which they’re designed.

Code Coverage
A way to identify which lines of code are exercised by a set of unit tests, and which aren’t. Code coverage helps you identify sections of code that are untested and therefore at greatest risk of containing a bug or introducing a regression in the future.

Component, Metadata
A component is an instance of a metadata type in the Metadata API. For example, CustomObject is a metadata type for custom objects, and the MyCustomObject__c component is an instance of a custom object. A component is described in an XML file and it can be deployed or retrieved using the Metadata API, or tools built on top of it, such as the Salesforce extensions for Visual Studio Code or the Ant Migration Tool.

Component, Visualforce
Something that can be added to a Visualforce page with a set of tags, for example, `<apex:detail>`. Visualforce includes a number of standard components, or you can create your own custom components.

Component Reference, Visualforce
A description of the standard and custom Visualforce components that are available in your organization. You can access the component library from the development footer of any Visualforce page or the Visualforce Developer’s Guide.
Composite App
An app that combines native platform functionality with one or more external Web services, such as Yahoo! Maps. Composite apps allow for more flexibility and integration with other services, but can require running and managing external code. See also Client App and Native App.

Controller, Visualforce
An Apex class that provides a Visualforce page with the data and business logic it must run. Visualforce pages can use the standard controllers that come by default with every standard or custom object, or they can use custom controllers.

Controller Extension
A controller extension is an Apex class that extends the functionality of a standard or custom controller.

Cookie
Client-specific data used by some Web applications to store user and session-specific information. Salesforce issues a session "cookie" only to record encrypted authentication information for the duration of a specific session.

Custom App
See App.

Custom Controller
A custom controller is an Apex class that implements all of the logic for a page without using a standard controller. Use custom controllers when you want your Visualforce page to run entirely in system mode, which doesn’t enforce the permissions and field-level security of the current user.

Custom Field
A field that can be added in addition to the standard fields to customize Salesforce for your organization’s needs.

Custom Links
Custom links are URLs defined by administrators to integrate your Salesforce data with external websites and back-office systems. Formerly known as Web links.

Custom Object
Custom records that allow you to store information unique to your organization.

Custom Settings
Custom settings are similar to custom objects and enable application developers to create custom sets of data, as well as create and associate custom data for an organization, profile, or specific user. All custom settings data is exposed in the application cache, which enables efficient access without the cost of repeated queries to the database. This data can then be used by formula fields, validation rules, flows, Apex, and SOAP API.
See also Hierarchy Custom Settings and List Custom Settings.

D

Database
An organized collection of information. The underlying architecture of the Lightning Platform includes a database where your data is stored.

Database Table
A list of information, presented with rows and columns, about the person, thing, or concept you want to track. See also Object.

Data Loader
A Lightning Platform tool used to import and export data from your Salesforce organization.

Data Manipulation Language (DML)
An Apex method or operation that inserts, updates, or deletes records.
Data State
The structure of data in an object at a particular point in time.

Date Literal
A keyword in a SOQL or SOSL query that represents a relative range of time such as last month or next year.

Decimal Places
Parameter for number, currency, and percent custom fields that indicates the total number of digits you can enter to the right of a decimal point, for example, 4.98 for an entry of 2. The system rounds the decimal numbers you enter, if necessary. For example, if you enter 4.986 in a field with Decimal Places of 2, the number rounds to 4.99. Salesforce uses the round half-up rounding algorithm. Half-way values are always rounded up. For example, 1.45 is rounded to 1.5. –1.45 is rounded to –1.5.

Dependency
A relationship where one object’s existence depends on that of another. There are a number of different kinds of dependencies including mandatory fields, dependent objects (parent-child), file inclusion (referenced images, for example), and ordering dependencies (when one object must be deployed before another object).

Dependent Field
Any custom picklist or multi-select picklist field that displays available values based on the value selected in its corresponding controlling field.

Deploy
To move functionality from an inactive state to active. For example, when developing new features in the Salesforce user interface, you must select the “Deployed” option to make the functionality visible to other users.

The process by which an application or other functionality is moved from development to production.

To move metadata components from a local file system to a Salesforce organization.

For installed apps, deployment makes any custom objects in the app available to users in your organization. Before a custom object is deployed, it’s only available to administrators and any users with the “Customize Application” permission.

Deprecated Component
Developers can refine the functionality in a managed package over time as the requirements evolve, such as redesign some of the components in the managed package. Developers can’t delete some components in a Managed - Released package, but they can deprecate a component in a later package version so that new subscribers don’t receive the component, while the component continues to function for existing subscribers and API integrations.

Detail
A page that displays information about a single object record. The detail page of a record allows you to view the information, whereas the edit page allows you to modify it.

A term used in reports to distinguish between summary information and inclusion of all column data for all information in a report. You can toggle the Show Details/Hide Details button to view and hide report detail information.

Developer Edition
A free, fully functional Salesforce organization designed for developers to extend, integrate, and develop with the Lightning Platform. Developer Edition accounts are available on developer.salesforce.com.

Salesforce Developers
The Salesforce Developers website at developer.salesforce.com provides a full range of resources for platform developers, including sample code, toolkits, an online developer community, and the ability to obtain limited Lightning Platform environments.

Development Environment
A Salesforce organization where you can make configuration changes that don’t affect users on the production organization. There are two kinds of development environments, sandboxes and Developer Edition organizations.
Email Alert
Email alerts are actions that send emails, using a specified email template, to specified recipients.

Email Template
A form email that communicates a standard message, such as a welcome letter to new employees or an acknowledgment that a customer service request has been received. Email templates can be personalized with merge fields, and can be written in text, HTML, or custom format.

Note: Lightning email templates aren't packageable.

Enterprise Edition
A Salesforce edition designed for larger, more complex businesses.

Enterprise WSDL
A strongly typed WSDL for customers who want to build an integration with their Salesforce organization only, or for partners who are using tools like Tibco or webMethods to build integrations that require strong typecasting. The downside of the Enterprise WSDL is that it only works with the schema of a single Salesforce organization because it's bound to all of the unique objects and fields that exist in that organization's data model.

Entity Relationship Diagram (ERD)
A data modeling tool that helps you organize your data into entities (or objects, as they're called in the Lightning Platform) and define the relationships between them. ERDs for key Salesforce objects are published in the Salesforce Object Reference.

Enumeration Field
An enumeration is the WSDL equivalent of a picklist field. The valid values of the field are restricted to a strict set of possible values, all having the same data type.

Facet
A child of another Visualforce component that allows you to override an area of the rendered parent with the contents of the facet.

Field
A part of an object that holds a specific piece of information, such as a text or currency value.

Field Dependency
A filter that allows you to change the contents of a picklist based on the value of another field.

Field-Level Security
Settings that determine whether fields are hidden, visible, read only, or editable for users. Available in Professional, Enterprise, Unlimited, Performance, and Developer Editions.

Foreign Key
A field whose value is the same as the primary key of another table. You can think of a foreign key as a copy of a primary key from another table. A relationship is made between two tables by matching the values of the foreign key in one table with the values of the primary key in another.

Getter Methods
Methods that enable developers to display database and other computed values in page markup. Methods that return values. See also Setter Methods.
Global Variable
A special merge field that you can use to reference data in your organization.
A method access modifier for any method that must be referenced outside of the application, either in SOAP API or by other Apex code.

Governor Limits
Apex execution limits that prevent developers who write inefficient code from monopolizing the resources of other Salesforce users.

Gregorian Year
A calendar based on a 12-month structure used throughout much of the world.

Hierarchy Custom Settings
A type of custom setting that uses a built-in hierarchical logic that lets you “personalize” settings for specific profiles or users. The hierarchy logic checks the organization, profile, and user settings for the current user and returns the most specific, or “lowest,” value. In the hierarchy, settings for an organization are overridden by profile settings, which, in turn, are overridden by user settings.

HTTP Debugger
An application that can be used to identify and inspect SOAP requests that are sent from the AJAX Toolkit. They behave as proxy servers running on your local machine and allow you to inspect and author individual requests.

ID
See Salesforce Record ID.

IdeaExchange
Salesforce’s always-on feedback platform that connects the Trailblazer Community with Salesforce product managers. It’s the go-to place to post ideas and contribute feedback about how to improve products and experiences. Visit IdeaExchange at ideas.salesforce.com.

Import Wizard
A tool for importing data into your Salesforce organization, accessible from Setup.

Instance
The cluster of software and hardware represented as a single logical server that hosts an organization’s data and runs their applications. The Lightning Platform runs on multiple instances, but data for any single organization is always stored on a single instance.

Integrated Development Environment (IDE)
A software application that provides comprehensive facilities for software developers including a source code editor, testing and debugging tools, and integration with source code control systems.

Integration User
A Salesforce user defined solely for client apps or integrations. Also referred to as the logged-in user in a SOAP API context.

ISO Code
The International Organization for Standardization country code, which represents each country by two letters.
Junction Object

A custom object with two master-detail relationships. Using a custom junction object, you can model a "many-to-many" relationship between two objects. For example, you create a custom object called "Bug" that relates to the standard case object such that a bug could be related to multiple cases and a case could also be related to multiple bugs.

Length

Parameter for custom text fields that specifies the maximum number of characters (up to 255) that a user can enter in the field.

Parameter for number, currency, and percent fields that specifies the number of digits you can enter to the left of the decimal point, for example, 123.98 for an entry of 3.

Lightning Platform

The Salesforce platform for building applications in the cloud. Lightning Platform combines a powerful user interface, operating system, and database to allow you to customize and deploy applications in the cloud for your entire enterprise.

List Custom Settings

A type of custom setting that provides a reusable set of static data that can be accessed across your organization. If you use a particular set of data frequently within your application, putting that data in a list custom setting streamlines access to it. Data in list settings doesn’t vary with profile or user, but is available organization-wide. Examples of list data include two-letter state abbreviations, international dialing prefixes, and catalog numbers for products. Because the data is cached, access is low-cost and efficient: you don’t have to use SOQL queries that count against your governor limits.

List View

A list display of items (for example, accounts or contacts) based on specific criteria. Salesforce provides some predefined views.

In the Agent console, the list view is the top frame that displays a list view of records based on specific criteria. The list views you can select to display in the console are the same list views defined on the tabs of other objects. You can’t create a list view within the console.

Local Name

The value stored for the field in the user’s or account’s language. The local name for a field is associated with the standard name for that field.

Locale

The country or geographic region in which the user is located. The setting affects the format of date and number fields, for example, dates in the English (United States) locale display as 06/30/2000 and as 30/06/2000 in the English (United Kingdom) locale.

In Professional, Enterprise, Unlimited, Performance, and Developer Edition organizations, a user’s individual Locale setting overrides the organization’s Default Locale setting. In Personal and Group Editions, the organization-level locale field is called Locale, not Default Locale.

Long Text Area

Data type of custom field that allows entry of up to 32,000 characters on separate lines.

Lookup Relationship

A relationship between two records so you can associate records with each other. For example, cases have a lookup relationship with assets that lets you associate a particular asset with a case. On one side of the relationship, a lookup field allows users to click a lookup icon and select another record from a window. On the associated record, you can then display a related list to show all of the records that have been linked to it. If a lookup field references a record that has been deleted, by default Salesforce clears the lookup field. Alternatively, you can prevent records from being deleted if they’re in a lookup relationship.
Managed Package
A collection of application components that is posted as a unit on AppExchange and associated with a namespace and possibly a License Management Organization. To support upgrades, a package must be managed. An organization can create a single managed package that can be downloaded and installed by many different organizations. Managed packages differ from unmanaged packages by having some locked components, allowing the managed package to be upgraded later. Unmanaged packages don’t include locked components and can’t be upgraded. In addition, managed packages obfuscate certain components (like Apex) on subscribing organizations to protect the intellectual property of the developer.

Manual Sharing
Record-level access rules that allow record owners to give read and edit permissions to other users who don’t have access to the record any other way.

Many-to-Many Relationship
A relationship where each side of the relationship can have many children on the other side. Many-to-many relationships are implemented through the use of junction objects.

Master-Detail Relationship
A relationship between two different types of records that associates the records with each other. For example, accounts have a master-detail relationship with opportunities. This type of relationship affects record deletion, security, and makes the lookup relationship field required on the page layout.

Metadata
Information about the structure, appearance, and functionality of an organization and any of its parts. Lightning Platform uses XML to describe metadata.

Metadata-Driven Development
An app development model that allows apps to be defined as declarative “blueprints,” with no code required. Apps built on the platform—their data models, objects, forms, workflows, and more—are defined by metadata.

Metadata WSDL
A WSDL for users who want to use the Lightning Platform Metadata API calls.

Multitenancy
An application model where all users and apps share a single, common infrastructure and code base.

MVC (Model-View-Controller)
A design paradigm that deconstructs applications into components that represent data (the model), ways of displaying that data in a user interface (the view), and ways of manipulating that data with business logic (the controller).

Namespace
In a packaging context, a one- to 15-character alphanumeric identifier that distinguishes your package and its contents from packages of other developers on AppExchange, similar to a domain name. Salesforce automatically prepends your namespace prefix, followed by two underscores (“__”), to all unique component names in your Salesforce organization.

Native App
An app that is built exclusively with setup (metadata) configuration on Lightning Platform. Native apps don’t require any external services or infrastructure.
Object
An object allows you to store information in your Salesforce organization. The object is the overall definition of the type of information you're storing. For example, the case object lets you store information regarding customer inquiries. For each object, your organization has multiple records that store the information about specific instances of that type of data. For example, you can have a case record to store the information about Joe Smith’s training inquiry and another case record to store the information about Mary Johnson's configuration issue.

Object-Level Help
Custom help text that you can provide for any custom object. It displays on custom object record home (overview), detail, and edit pages, as well as list views and related lists.

Object-Level Security
Settings that allow an administrator to hide whole objects from users so that they don't know that type of data exists. Object-level security is specified with object permissions.

One-to-Many Relationship
A relationship in which a single object is related to many other objects. For example, an account can have one or more related contacts.

Organization
A deployment of Salesforce with a defined set of licensed users. An organization is the virtual space provided to an individual customer of Salesforce. Your organization includes all of your data and applications, and is separate from all other organizations.

Organization-Wide Defaults
Settings that allow you to specify the baseline level of data access that a user has in your organization. For example, you can set organization-wide defaults so that any user can see any record of a particular object that is enabled via their object permissions, but they need extra permissions to edit one.

Outbound Call
Any call that originates from a user to a number outside of a call center in Salesforce CRM Call Center.

Outbound Message
An outbound message sends information to a designated endpoint, like an external service. Outbound messages are configured from Setup. You must configure the external endpoint and create a listener for the messages using SOAP API.

Owner
Individual user to which a record (for example, a contact or case) is assigned.

P

PaaS
See Platform as a Service.

Package
A group of Lightning Platform components and applications that are made available to other organizations through AppExchange. You use packages to bundle an app along with any related components so that you can upload them to AppExchange together.

Package Dependency
This dependency is created when one component references another component, permission, or preference that is required for the component to be valid. Components can include but aren’t limited to:

- Standard or custom fields
- Standard or custom objects
- Visualforce pages
• Apex code

Permissions and preferences can include but aren’t limited to:
• Divisions
• Multicurrency
• Record types

Package Installation
Installation incorporates the contents of a package into your Salesforce organization. A package on AppExchange can include an app, a component, or a combination of the two. After you install a package, you can deploy components in the package to make it generally available to the users in your organization.

Package Version
A package version is a number that identifies the set of components uploaded in a package. The version number has the format majorNumber.minorNumber.patchNumber (for example, 2.1.3). The major and minor numbers increase to a chosen value during every major release. The patchNumber is generated and updated only for a patch release.

Unmanaged packages aren’t upgradeable, so each package version is simply a set of components for distribution. A package version has more significance for managed packages. Packages can exhibit different behavior for different versions. Publishers can use package versions to evolve the components in their managed packages gracefully by releasing subsequent package versions without breaking existing customer integrations using the package. See also Patch and Patch Development Organization.

Partner WSDL
A loosely typed WSDL for customers, partners, and ISVs who want to build an integration or an AppExchange app that can work across multiple Salesforce organizations. With this WSDL, the developer is responsible for marshaling data in the correct object representation, which typically involves editing the XML. However, the developer is also freed from being dependent on any particular data model or Salesforce organization. Contrast to the Enterprise WSDL, which is strongly typed.

Patch
A patch enables a developer to change the functionality of existing components in a managed package, while ensuring subscribing organizations that there are no visible behavior changes to the package. For example, you can add new variables or change the body of an Apex class, but you can’t add, deprecate, or remove any of its methods. Patches are tracked by a patchNumber appended to every package version. See also Patch Development Organization and Package Version.

Patch Development Organization
The organization where patch versions are developed, maintained, and uploaded. Patch development organizations are created automatically for a developer organization when they request to create a patch. See also Patch and Package Version.

Personal Edition
Product designed for individual sales representatives and single users.

Platform as a Service (PaaS)
An environment where developers use programming tools offered by a service provider to create applications and deploy them in a cloud. The application is hosted as a service and provided to customers via the Internet. The PaaS vendor provides an API for creating and extending specialized applications. The PaaS vendor also takes responsibility for the daily maintenance, operation, and support of the deployed application and each customer’s data. The service alleviates the need for programmers to install, configure, and maintain the applications on their own hardware, software, and related IT resources. Services can be delivered using the PaaS environment to any market segment.

Platform Edition
A Salesforce edition based on Enterprise, Unlimited, or Performance Edition that doesn’t include any of the standard Salesforce apps, such as Sales or Service & Support.
Primary Key
A relational database concept. Each table in a relational database has a field in which the data value uniquely identifies the record. This field is called the primary key. The relationship is made between two tables by matching the values of the foreign key in one table with the values of the primary key in another.

Production Organization
A Salesforce organization that has live users accessing data.

Professional Edition
A Salesforce edition designed for businesses who need full-featured CRM functionality.

Prototype
The classes, methods, and variables that are available to other Apex code.

Query Locator
A parameter returned from the `query()` or `queryMore()` API call that specifies the index of the last result record that was returned.

Query String Parameter
A name-value pair that’s included in a URL, typically after a ‘?’ character. For example:

https://yourInstance.salesforce.com/001/e?name=value

Record
A single instance of a Salesforce object. For example, “John Jones” can be the name of a contact record.

Record ID
The unique identifier for each record.

Record-Level Security
A method of controlling data in which you can allow a particular user to view and edit an object, but then restrict the records that the user is allowed to see.

Record Locking
Record locking prevents users from editing a record, regardless of field-level security or sharing settings. By default, Salesforce locks records that are pending approval. Only admins can edit locked records.

Record Name
A standard field on all Salesforce objects. Whenever a record name is displayed in a Lightning Platform application, the value is represented as a link to a detail view of the record. A record name can be either free-form text or an autonumber field. Record Name doesn’t have to be a unique value.

Recycle Bin
A page that lets you view and restore deleted information. Access the Recycle Bin either by using the link in the sidebar in Salesforce Classic or from the App Launcher in Lightning Experience.

Relationship
A connection between two objects, used to create related lists in page layouts and detail levels in reports. Matching values in a specified field in both objects are used to link related data; for example, if one object stores data about companies and another object stores data about people, a relationship allows you to find out which people work at the company.
Relationship Query
In a SOQL context, a query that traverses the relationships between objects to identify and return results. Parent-to-child and child-to-parent syntax differs in SOQL queries.

Role Hierarchy
A record-level security setting that defines different levels of users such that users at higher levels can view and edit information owned by or shared with users beneath them in the role hierarchy, regardless of the organization-wide sharing model settings.

Roll-Up Summary Field
A field type that automatically provides aggregate values from child records in a master-detail relationship.

Running User
Each dashboard has a running user, whose security settings determine which data to display in a dashboard. If the running user is a specific user, all dashboard viewers see data based on the security settings of that user—regardless of their own personal security settings. For dynamic dashboards, you can set the running user to be the logged-in user, so that each user sees the dashboard according to their own access level.

S
SaaS
See Software as a Service (SaaS).

S-Control

Note: S-controls have been superseded by Visualforce pages. After March 2010 organizations that have never created s-controls, as well as new organizations, won’t be allowed to create them. Existing s-controls remain unaffected, and can still be edited.

Custom Web content for use in custom links. Custom s-controls can contain any type of content that you can display in a browser, for example a Java applet, an Active-X control, an Excel file, or a custom HTML Web form.

Salesforce Certificate and Key Pair
Salesforce certificates and key pairs are used for signatures that verify a request is coming from your organization. They’re used for authenticated SSL communications with an external website, or when using your organization as an Identity Provider. You only must generate a Salesforce certificate and key pair if you’re working with an external website that wants verification that a request is coming from a Salesforce organization.

Salesforce Connect
Salesforce Connect provides access to data that’s stored outside the Salesforce org, such as data in an enterprise resource planning (ERP) system and records in another org. Salesforce Connect represents the data in external objects and accesses the external data in real time via Web service callouts to external data sources.

Salesforce Extensions for Visual Studio Code
The Salesforce extension pack for Visual Studio Code includes tools for developing on the Salesforce platform in the lightweight, extensible VS Code editor. These tools provide features for working with development orgs (scratch orgs, sandboxes, and DE orgs), Apex, Aura components, and Visualforce.

Salesforce Record ID
A unique 15-character or 18-character alphanumeric string that identifies a single record in Salesforce.

Salesforce Service Oriented Architecture
A powerful capability of Lightning Platform that allows you to make calls to external Web services from within Apex.

Sandbox
A nearly identical copy of a Salesforce production organization for development, testing, and training. The content and size of a sandbox varies depending on the type of sandbox and the edition of the production organization associated with the sandbox.
Semi-Join
A semi-join is a subquery on another object in an IN clause in a SOQL query. You can use semi-joins to create advanced queries, such as getting all contacts for accounts that have an opportunity with a particular record type. See also Anti-Join.

Session ID
An authentication token that is returned when a user successfully logs in to Salesforce. The Session ID prevents a user from having to log in again every time they want to perform another action in Salesforce. Different from a record ID or Salesforce ID, which are terms for the unique ID of a Salesforce record.

Session Timeout
The time after login before a user is automatically logged out. Sessions expire automatically after a predetermined length of inactivity, which can be configured in Salesforce from Setup by clicking Security Controls. The default is 120 minutes (two hours). The inactivity timer is reset to zero if a user takes an action in the web interface or makes an API call.

Setter Methods
Methods that assign values. See also Getter Methods.

Setup
A menu where administrators can customize and define organization settings and Lightning Platform apps. Depending on your organization’s user interface settings, Setup can be a link in the user interface header or in the dropdown list under your name.

Sharing
Allowing other users to view or edit information you own. There are different ways to share data:

- Sharing Model—defines the default organization-wide access levels that users have to each other’s information and whether to use the hierarchies when determining access to data.
- Role Hierarchy—defines different levels of users such that users at higher levels can view and edit information owned by or shared with users beneath them in the role hierarchy, regardless of the organization-wide sharing model settings.
- Sharing Rules—allow an administrator to specify that all information created by users within a given group or role is automatically shared to the members of another group or role.
- Manual Sharing—allows individual users to share records with other users or groups.
- Apex-Managed Sharing—enables developers to programmatically manipulate sharing to support their application’s behavior. See Apex-Managed Sharing.

Sharing Model
Behavior defined by your administrator that determines default access by users to different types of records.

Sharing Rule
Type of default sharing created by administrators. Allows users in a specified group or role to have access to all information created by users within a given group or role.

Sites
Salesforce Sites enables you to create public websites and applications that are directly integrated with your Salesforce organization—without requiring users to log in with a username and password.

SOAP (Simple Object Access Protocol)
A protocol that defines a uniform way of passing XML-encoded data.

SOAP API
A SOAP-based Web services application programming interface that provides access to your Salesforce organization’s information.

sObject
The abstract or parent object for all objects that can be stored in the Lightning Platform.

Software as a Service (SaaS)
A delivery model where a software application is hosted as a service and provided to customers via the Internet. The SaaS vendor takes responsibility for the daily maintenance, operation, and support of the application and each customer’s data. The service
alleviates the need for customers to install, configure, and maintain applications with their own hardware, software, and related IT resources. Services can be delivered using the SaaS model to any market segment.

**SOQL (Salesforce Object Query Language)**
A query language that allows you to construct simple but powerful query strings and to specify the criteria that selects data from the Lightning Platform database.

**SOSL (Salesforce Object Search Language)**
A query language that allows you to perform text-based searches using the Lightning Platform API.

**Standard Object**
A built-in object included with the Lightning Platform. You can also build custom objects to store information that is unique to your app.

**System Log**
Part of the Developer Console, a separate window console that can be used for debugging code snippets. Enter the code you want to test at the bottom of the window and click Execute. The body of the System Log displays system resource information, such as how long a line took to execute or how many database calls were made. If the code didn’t run to completion, the console also displays debugging information.

**T**

**Tag**
In Salesforce, a word or short phrases that users can associate with most records to describe and organize their data in a personalized way. Administrators can enable tags for accounts, activities, assets, campaigns, cases, contacts, contracts, dashboards, documents, events, leads, notes, opportunities, reports, solutions, tasks, and any custom objects (except relationship group members). Tags can also be accessed through SOAP API.

**Test Case Coverage**
Test cases are the expected real-world scenarios in which your code is used. Test cases aren’t actual unit tests, but are documents that specify what your unit tests do. High test case coverage means that most or all the real-world scenarios you have identified are implemented as unit tests. See also Code Coverage and Unit Test.

**Test Method**
An Apex class method that verifies whether a particular piece of code is working properly. Test methods take no arguments, commit no data to the database, and can be executed by the `runTests()` system method either through the command line or in an Apex IDE, such as the Salesforce extensions for Visual Studio Code.

**Test Organization**
See Sandbox.

**Transaction, Apex**
An Apex transaction represents a set of operations that are executed as a single unit. All DML operations in a transaction either complete successfully, or if an error occurs in one operation, the entire transaction is rolled back and no data is committed to the database. The boundary of a transaction can be a trigger, a class method, an anonymous block of code, a Visualforce page, or a custom Web service method.

**Trigger**
A piece of Apex that executes before or after records of a particular type are inserted, updated, or deleted from the database. Every trigger runs with a set of context variables that provide access to the records that caused the trigger to fire, and all triggers run in bulk mode—that is, they process several records at once, rather than just one record at a time.

**Trigger Context Variable**
Default variables that provide access to information about the trigger and the records that caused it to fire.
Unit Test
A unit is the smallest testable part of an application, usually a method. A unit test operates on that piece of code to make sure it works correctly. See also Test Method.

Unlimited Edition
Unlimited Edition is Salesforce's solution for maximizing your success and extending that success across the entire enterprise through the Lightning Platform.

Unmanageable Package
A package that can't be upgraded or controlled by its developer.

URL (Uniform Resource Locator)
The global address of a website, document, or other resource on the Internet. For example, https://salesforce.com.

User Acceptance Testing (UAT)
A process used to confirm that the functionality meets the planned requirements. UAT is one of the final stages before deployment to production.

Validation Rule
A rule that prevents a record from being saved if it doesn't meet the standards that are specified.

Version
A number value that indicates the release of an item. Items that can have a version include API objects, fields, and calls; Apex classes and triggers; and Visualforce pages and components.

View
The user interface in the Model-View-Controller model, defined by Visualforce.

View State
Where the information necessary to maintain the state of the database between requests is saved.

Visualforce
A simple, tag-based markup language that allows developers to easily define custom pages and components for apps built on the platform. Each tag corresponds to a coarse or fine-grained component, such as a section of a page, a related list, or a field. The components can either be controlled by the same logic that is used in standard Salesforce pages, or developers can associate their own logic with a controller written in Apex.

Visualforce Controller
See Controller, Visualforce.

Visualforce Lifecycle
The stages of execution of a Visualforce page, including how the page is created and destroyed during a user session.

Visualforce Page
A web page created using Visualforce. Typically, Visualforce pages present information relevant to your organization, but they can also modify or capture data. They can be rendered in several ways, such as a PDF document or an email attachment, and can be associated with a CSS style.

Web Service
A mechanism by which two applications can easily exchange data over the Internet, even if they run on different platforms, are written in different languages, or are geographically remote from each other.
WebService Method
An Apex class method or variable that external systems can use, like a mash-up with a third-party application. Web service methods must be defined in a global class.

Web Services API
Term describing the original Salesforce Platform web services application programming interface (API) that provides access to your Salesforce org’s information. See relevant developer guides for SOAP, REST, or Bulk APIs of interest.

Automated Actions
Automated actions, such as email alerts, tasks, field updates, and outbound messages, can be triggered by a process, workflow rule, approval process, or milestone.

Wrapper Class
A class that abstracts common functions such as logging in, managing sessions, and querying and batching records. A wrapper class makes an integration more straightforward to develop and maintain, keeps program logic in one place, and affords easy reuse across components. Examples of wrapper classes in Salesforce include the AJAX Toolkit, which is a JavaScript wrapper around the Salesforce SOAP API, wrapper classes such as Critical Section in the CTI Adapter for Salesforce CRM Call Center, or wrapper classes created as part of a client integration application that accesses Salesforce using SOAP API.

WSDL (Web Services Description Language) File
An XML file that describes the format of messages you send and receive from a Web service. Your development environment’s SOAP client uses the Salesforce Enterprise WSDL or Partner WSDL to communicate with Salesforce using SOAP API.

XML (Extensible Markup Language)
A markup language that enables the sharing and transportation of structured data. All Lightning Platform components that are retrieved or deployed through the Metadata API are represented by XML definitions.

X

No Glossary items for this entry.

Y

No Glossary items for this entry.

Z

No Glossary items for this entry.